Bent functions, SDP designs and their automorphism groups

A thesis submitted to the
University of London for the
degree of Doctor of Philosophy

Thomas David Bending

Queen Mary and Westfield College,
November 1993

Bent functions, SDP designs and their automorphism groups

Abstract

In a 1976 paper Rothaus coined the term "bent" to describe a function f from a vector space $V(n, 2)$ to $\mathbb{F}_2$ with the property that the Fourier coefficients of $(-1)^f$ have unit magnitude. Such a function has the maximum possible distance from the set of linear functions, hence the name, and has useful correlation properties. These lead to various applications to coding theory and cryptography, some of which are described. A standard notion of the equivalence of two bent functions is discussed and related to the coding theory setting.

Two constructions mentioned by Rothaus and generalised by Maiorana are described. A further generalisation of one of these, involving sets of bent functions on direct summands of the original vector space, is described and proved. Various methods including computer searches are used to find appropriate sets of bent functions and hence many new equivalence classes of bent functions of 8 variables. Equivalence class invariants are used to show that most of these classes cannot be constructed by the earlier methods. Some bounds on numbers of bent functions are discussed.

A 2-design is said to have the symmetric difference property (SDP) if the symmetric difference of any three blocks is either a block or the complement of a block — such a design is very close to being a 3-design. All SDP designs are induced by bent functions, and conversely. Work on the automorphism groups of various SDP designs involving computer algebra is described. An SDP design on 256 points with trivial automorphism group is noted.

Some connections with strongly-regular graphs are discussed. An infinite class of pseudo-geometric strongly-regular graphs induced by bent functions is noted, and bent functions which are their own Fourier transform duals are investigated. Finally, some open problems and ideas for future work are described.

Acknowledgements

I am very grateful to Professor P. J. Cameron, my supervisor, for his suggestion of this topic and for his invaluable help and advice throughout my work on it. His lightning explanations and endless supply of examples have been an inspiration.

I would like to thank colleagues in the Queen Mary and Westfield Combinatorics Study Group and elsewhere for a number of enjoyable and useful discussions, especially Simon McLeish and Dima Fon-Der-Flaass. In particular I am grateful to Ken Paterson for drawing my attention to the cryptography side of the subject, and pointing me at some of its literature.

This work was supported by Science and Engineering Research Council Research Studentship 90317694, for which I am very grateful.

Contents

1. Introduction **9**

Fourier transforms 9

Bent functions - definitions and basic properties 13

Examples of bent functions 16

An alternative characterisation 17

2. Motivation from coding theory **20**

Reed-Muller codes 20

The code induced by a bent function 21

Weight enumerators 22

Other applications to coding theory and cryptography 24

3. Equivalent bent functions **25**

Equivalences of codes induced by bent functions 28

Equivalence classes of bent functions 28

Quadratic forms 30

More about equivalence classes of bent functions 33

4. Two bent function constructions **35**

The Maiorana construction 35

Duality 37

Duality and equivalence 39

5. The four-function construction **41**

Inducing lower-order bent functions 41

The four-function construction 45

Another way of representing the four-function construction 46

6. Generalisation of the four-function construction **50**

Ways of forming suitable matrices 55

Performance of the various methods for $n = 8$ 62

7. Comments on the direct-summand construction **63**

"Jordan-Hölder" conjectures 63

Estimating the number of bent functions — upper bounds 65

Estimating the number of bent functions — lower bounds 66

Estimating the number of equivalence classes 67

8. Equivalence class invariants **68**
The stabiliser graph 75
Other invariants 76

9. Designs **77**
Inducing a design from a bent function 78

10. Automorphisms of designs **83**
Finding the addition designs' automorphism groups 87
Some remarks on the addition designs' automorphism groups 90

11. More about designs **92**
A second design induced by a bent function 92
Derived and residual designs 94

12. A connection with strongly-regular graphs **97**
A class of structures respected by equivalences 101

13. A connection with partial geometries **104**

14. Self-dual bent functions **116**

15. Open problems and conclusions **120**
Typical bent functions 120
Kerdock codes 121
Generalised bent functions 122
Relationships between structures 122

A. Classes of bent functions with $n = 8$ **124**

B. Stabiliser space dimensions $\underline{d}(f)$ **137**

C. Design automorphism and orbit details **141**

References **143**

1. Introduction

The term "bent" was coined by Rothaus [35] to describe a class of functions from a vector space $V(n, 2)$ to its ground field $\mathbb{F}_2$. These are characterised in terms of a generalisation of the classical Fourier transform — if we convert from $\mathbb{F}_2$-valued functions to complex-valued functions with range ± 1 then bent functions are those whose transforms are also confined to this range.

Bent functions were originally considered in connection with difference sets, but turn out to have various properties which are important in coding theory and cryptography. They also have connections with various combinatorial structures, in particular SDP designs (2-designs which are very nearly 3-designs).

In this thesis we define bent functions and discuss various of their properties, and then consider a variety of constructions. The main result is a generalisation of one of these which is used to produce many new classes of bent functions of 8 variables — we show that this construction is indeed strictly stronger than the earlier ones. We describe the connection with SDP designs and use it to study the automorphism groups of the designs, which typically seem to be rather smaller than is suggested by small examples. We examine connections with strongly-regular graphs, partial geometries and other structures, and finally outline some open problems and ideas for future work.

Fourier transforms

The classical notion of the Fourier transform of a periodic complex-valued function on $\mathbb{R}$ involves encoding the function in terms of the coefficients of a basis of simpler functions. This idea can be applied in a much more general setting by considering matrix representations of groups. Much of the following discussion is based on Kirillov [20].

Given a group G and a field K, a *(matrix) representation of G over K* is a homomorphism $\theta : G \to \mathrm{GL}_d(K)$, where $\mathrm{GL}_d(K)$ is the *general linear group* of non-singular $d \times d$ matrices over K. $\mathrm{GL}_d(K)$ acts on a vector space $\mathcal{V}$ of dimension d over K. θ is said to be *reducible* if $\mathcal{V}$ can be written as a direct sum of vector spaces preserved setwise by the matrices $\theta(G)$, and *irreducible* otherwise.

Given a representation θ we can define its *character* χ by setting $\chi(g) = \mathrm{tr}(\theta(g))$ where $\mathrm{tr}(\bullet)$ denotes the trace function. The irreducible characters (those corresponding to irreducible representations) generate a semiring under pointwise addition and multiplication.

Although the theory can be developed quite generally, for our purposes we need

consider only the special case $K = \mathbb{C}$, the field of complex numbers. Also we will assume that G is a locally compact topological group (which is true if G is finite) and that G is abelian. In this case the irreducible characters are all 1-dimensional ($d = 1$). Thus a representation is just a homomorphism $G \to \mathbb{C}^\times$, and we can identify a representation with its character. A character χ is said to be *unitary* if for all $g \in G$ we have $|\chi(g)| = 1$ — note that this must be true for those g which have finite order.

With our assumptions the unitary irreducible characters form a group $\hat{G}$, with multiplication defined pointwise, and $\chi^{-1} = \bar{\chi}$, the complex conjugate of χ. $\hat{G}$ is called the *dual group* of G. $\mathbb{C}^\times$ is abelian hence so is $\hat{G}$, so we can go on to form the group $\hat{\hat{G}}$. In fact there is a natural map

$$\begin{array}{rccc} \natural: & G & \to & \hat{\hat{G}} \\ & g & \mapsto & (\chi \mapsto \chi(g)). \end{array}$$

Definition: If $\varphi : G \to \mathbb{C}$ then $\tilde{\varphi}$, the *Fourier transform of* φ, is defined by

$$\tilde{\varphi}(\chi) = \int_G \varphi(g)\chi(g)\, dg.$$

□

The point of these notions is the following result of Pontrjagin:

Proposition 1.1: *With the above notation*

(i) *If G is a locally compact topological group then $\natural$ is a isomorphism of topological groups.*

(ii) *With an appropriate normalisation $\varphi(g) = \int_{\hat{G}} \tilde{\varphi}(g)\bar{\chi}(g)\, d\chi$.* □

In other words the Fourier transform operation is self-inverse. Essentially this follows from orthogonality relations between the irreducible characters — we will see an example of this in Corollary 1.4 and use it in Proposition 1.5.

If we consider functions on the real line $G = \mathbb{R}$ then we obtain the classical Fourier transform $\tilde{f}(\lambda) = \int_{-\infty}^{\infty} e^{i\lambda t} f(t)\, dt$. Similarly if we consider functions on the unit circle $G = \mathbb{R}/\mathbb{Z}$ (or equivalently periodic functions on $\mathbb{R}$) then we get $\tilde{f}(n) = \frac{1}{2\pi}\int_0^{2\pi} e^{in\alpha} f(\alpha)\, d\alpha$, where $\{\tilde{f}(n) : n \in \mathbb{Z}\}$ are the classical Fourier coefficients.

However, if G is a finite (abelian) group we can define the Fourier transform discretely as follows:

$$\tilde{\varphi}(\chi) = \frac{1}{\sqrt{|G|}} \sum_G \varphi(g)\chi(g)$$

where $\frac{1}{\sqrt{|G|}}$ is the normalisation factor. Proposition 1.1(ii) then becomes

$$\varphi(g) = \frac{1}{\sqrt{|G|}} \sum_G \tilde{\varphi}(g)\bar{\chi}(g).$$

We sometimes call $\{\tilde{\varphi}(g)\}$ the *Fourier coefficients* of φ.

Now we restrict further and consider $G = V(n, 2)$, a vector space of dimension n over $\mathbb{F}_2$ considered as an additive abelian group of order 2^n. Given a basis for V we define the usual dot-product

$$x.y = (x_1, \ldots, x_n).(y_1, \ldots, y_n) = \sum_{i=1}^{n} x_i y_i = x^T y.$$

Given a set of vectors $S \subseteq V$ we write $\langle S \rangle$ for the span of S, i.e. the space of linear combinations of vectors in S.

If we have a function $f : V \to \mathbb{F}_2$ then we can identify f with its *support*, the set of points on which it takes the value 1. For example we write $|f|$ to mean "the number of points on which f takes the value 1" — we also call this quantity the *weight* of f. Addition of such functions corresponds to taking the symmetric difference of their supports, and we sometimes refer to the weight of this symmetric difference as the *distance* between the two functions. We write $\mathrm{supp}(f)$ if we wish to emphasise that we are considering f as a set of points. We write $\mathbb{0}$ for the all-0 function, whose support is the empty set, and $\mathbb{1}$ for the all-1 function, whose support is V.

Given a vector $x \in V$, the set $x^\perp = \{y \in V : x.y = 0\}$ of vectors orthogonal to x forms a subspace of V. This is a hyperplane, i.e. has codimension 1, if $x \neq 0$. We call it the *hyperplane indexed by* x. Now if we have a linear function $l : V \to \mathbb{F}_2$ which is not identically 0 it is an easy check that the kernel of l is a subspace with only one coset, so it must be some hyperplane $x^\perp$. Thus the support of l is $\overline{x^\perp}$, although in fact we will usually write this as $x^{\bar{\perp}}$ for clarity. This in turn suggests the notation

$$x^{(a)} = \begin{cases} x^\perp & \text{if } a = 0 \\ x^{\bar{\perp}} & \text{if } a = 1 \end{cases}$$

so that $y \in x^{(a)} \iff x.y = a$. We call hyperplanes and their cosets *halfspaces*, and we often write $x^?$ to mean "either $x^\perp$ or $x^{\bar{\perp}}$". Note that $0^\perp = \mathbb{0}$ so $0^{\bar{\perp}} = \mathbb{1}$. We will sometimes use "$\hat{+}$" rather than "$+$" to emphasise that an addition is taking place in $\mathbb{F}_2$. For example

Lemma 1.2: $x^{(\sigma)} + y^{(\tau)} = (x + y)^{(\sigma \,\hat{+}\, \tau \,\hat{+}\, 1)}$.

Proof: $z \in x^{(\sigma)}$ iff $z.x \;\hat{+}\; \sigma = 0$, so $z \in x^{(\sigma)} + y^{(\tau)}$ iff $z.x \;\hat{+}\; \sigma \;\hat{+}\; z.y \;\hat{+}\; \tau = 1$ iff $z.(x + y) = \sigma \;\hat{+}\; \tau \;\hat{+}\; 1$ iff $z \in (x + y)^{(\sigma \,\hat{+}\, \tau \,\hat{+}\, 1)}$. □

So now let χ be an irreducible character of V, i.e. a homomorphism $V \to \mathbb{C}^\times$. χ is unitary since V is finite, and every non-zero $v \in V$ has order 2 as a group

element, so $\chi(v) = \pm 1$. Thus setting $\omega = -1$ we can define a function $c : V \to \mathbb{F}_2$ by $\chi(v) = \omega^{c(v)}$.

Now since χ is a homomorphism we have

$$\omega^{c(v+w)} = \chi(v+w) = \chi(v)\chi(w) = \omega^{c(v)}\omega^{c(w)} = \omega^{c(v) \,\hat{+}\, c(w)}$$

Thus we see that c is a linear function on V, so its support is $x^{\bar{\perp}}$ for some $x \in V$ (possibly with $x = 0$). Conversely any such linear function c induces a homomorphism $\chi \in \hat{V}$. Thus in this case we have a natural map between V and $\hat{V}$, with x corresponding to χ, where $\chi(v) = \omega^{x.v}$ (in fact this is true for a vector space over any field $\mathbb{F}_p$ of prime order if we take ω to be a pth root of unity). So we can make the following:

Definition: If $\varphi : V(n,2) \to \mathbb{C}$ then $\tilde{\varphi}$, the Fourier transform of φ, is given by

$$\tilde{\varphi}(x) = 2^{-n/2} \sum_{y \in V} \varphi(y)\omega^{y.x}.$$

□

Although we have Pontrjagin's result (Proposition 1.1) giving φ in terms of $\tilde{\varphi}$, in this case it is not too hard to check it directly:

Lemma 1.3: *If $y \in V(n,2)$ then* $\sum_{x \in V} \omega^{x.y} = \begin{cases} 0 & \text{if } y \neq 0 \\ 2^n & \text{if } y = 0. \end{cases}$

Proof: If $x \neq 0$ then $|x^{\perp}| = |x^{\bar{\perp}}|$ so as x runs over V $x.y$ will take the values 0 and 1 equally often, so $\omega^{x.y}$ will take the values $+1$ and -1 equally often, so the sum will be 0.

On the other hand if $x = 0$ then $\omega^{x.y}$ will always be 1 so the sum will be $|V| = 2^n$.

□

Corollary 1.4: $\sum_{x \in V} \omega^{x.y}\omega^{x.z} = \begin{cases} 0 & \text{if } y \neq z \\ 2^n & \text{if } y = z. \end{cases}$

Proof: $\omega^{x.y}\omega^{x.z} = \omega^{x.(y+z)}$ so use Lemma 1.3. □

This is the character orthogonality relation mentioned above, and it is vital in proving our special case of Proposition 1.1(ii):

Proposition 1.5: *If $\varphi : V(n,2) \to \mathbb{C}$ then*

$$\tilde{\varphi}(x) = 2^{-n/2} \sum_{y \in V} \varphi(y)\omega^{y.x} \qquad \Longleftrightarrow \qquad \varphi(y) = 2^{-n/2} \sum_{x \in V} \tilde{\varphi}(x)\omega^{x.y}.$$

Proof: Substituting for $\tilde{\varphi}(x)$ in the RHS we have

$$2^{-n/2}\sum_{x\in V}\tilde{\varphi}(x)\omega^{x.y} = 2^{-n/2}\sum_{x\in V}\left(2^{-n/2}\sum_{z\in V}\varphi(z)\omega^{z.x}\right)\omega^{x.y}$$
$$= 2^{-n}\sum_{z\in V}\varphi(z)\sum_{x\in V}\omega^{x.z}\omega^{x.y}.$$

Now by Corollary 1.4 the inner sum will be 0 unless $z = y$ in which case it will be 2^n. Thus the only non-zero term on the RHS is $2^{-n}2^n\varphi(y) = \varphi(y)$, as required. The other implication follows by symmetry. □

Corollary 1.6: *If $\varphi : V(n,2) \to \mathbb{C}$ then $\tilde{\tilde{\varphi}} = \varphi$, so the Fourier transform is self-inverse.* □

Bent functions — definitions and basic properties

Rothaus' definition of the Fourier transform of a function $f : V \to \mathbb{F}_2$ corresponds to ours if we take $\varphi = \omega^f$. His definition of *bent function* then corresponds to the following:

Definition: A function $f : V(n,2) \to \mathbb{F}_2$ is said to be *bent* if all the Fourier coefficients of ω^f are ± 1. □

Such functions are called bent because they are far from the set of linear functions — we will see more details of this and other coding theory applications in Chapter 2. Rothaus mentions an application to difference sets, described by McFarland [27], and proves the following useful fact:

Proposition 1.7: *If f is a bent function on $V(n,2)$ then n is even.*

Proof: If $\varphi = \omega^f$ then in calculating $\tilde{\varphi}(x)$ from the discrete definition the value of the sum is an integer, so if $\tilde{\varphi}(x)$ is to be an integer then n must be even. □

Because of this result we will often consider only even n, and in this case we write $n = 2m$.

Various other facts about bent functions, and some alternative characterisations of them, can be deduced immediately from the definition.

Lemma 1.8: *If $f : V(n,2) \to \mathbb{F}_2$ then $\sum_{x\in V}\omega^{f(x)} = 2^n - 2|f|$.*

Proof: Each side subtracts the number of points on which f takes the value 1 ($|f|$) from the number of points on which it takes the value 0 ($2^n - |f|$). □

Corollary 1.9: *If $f : V(n,2) \to \mathbb{F}_2$ and $\varphi = \omega^f$ then $2^{n/2}\tilde{\varphi}(x) = 2^n - 2|f + x^{\bar{\perp}}|$.*

Proof: Apply Lemma 1.8 to $f + x^{\bar{\perp}}$ and compare with the definition of $\tilde{\varphi}$. □

Proposition 1.10: *A function $f : V(2m, 2) \to \mathbb{F}_2$ is bent iff $|f + x^{\bar{\perp}}| = 2^{n-1} \pm 2^{m-1}$ for all $x \in V$.*

Proof: Using Corollary 1.9, f is bent iff $\tilde{\varphi}(x) = \pm 1$ for all $x \in V$ iff $2^n - 2|f + x^{\bar{\perp}}| = \pm 2^m$ for all $x \in V$ iff $|f + x^{\bar{\perp}}| = 2^{n-1} \pm 2^{m-1}$ for all $x \in V$. □

Corollary 1.11: *If f is bent then $|f| = 2^{n-1} \pm 2^{m-1}$.*

Proof: Take $x = 0$ in Proposition 1.10. □

This Corollary suggests the following:

Definition: If f is bent then we say that f is *light* (respectively *heavy*) if $|f| = 2^{n-1} - 2^{m-1}$ ($2^{n-1} + 2^{m-1}$). Similarly we say that $|f|$ is *low* (*high*). □

Lemma 1.12: $|A \triangle B| = |A| + |B| - 2|A \cap B|$.

Proof: Easy. □

Proposition 1.13: *$f : V \to \mathbb{F}_2$ is bent iff $|f| = 2^{n-1} \pm 2^{m-1}$ and for all non-zero $x \in V$*

$$|f \cap x^?| = \begin{cases} 2^{n-2} - 2^{m-1} \text{ or } 2^{n-2} & \text{if } f \text{ is light} \\ 2^{n-2} + 2^{m-1} \text{ or } 2^{n-2} & \text{if } f \text{ is heavy.} \end{cases}$$

Proof: Straightforward calculation using Proposition 1.10 and Lemma 1.12. □

It will often be useful to regard a function on V as a polynomial in the co-ordinates of its argument (with respect to some basis for V). We can then talk about the degree of a function, meaning the degree of any polynomial representation of it. Note that if $|K| = q$ then $x^q = x$ for all $x \in K$, so we can always eliminate factors of x^q from our polynomials before calculating the degree. In particular if $K = \mathbb{F}_2$ each co-ordinate only appears in a given term once — we never have terms like ${x_1}^2 x_2$.

With the usual addition the set of all functions from V to K forms a vector space. For $1 \leqslant r \leqslant n$ the functions of degree at most r form a subspace, the rth Reed-Muller code $\mathrm{RM}_r(n)$ or just RM_r. The functions of the form $x^?$, the halfspaces and constant functions, are exactly those in RM_1.

We often need to find the polynomial representation of a function when we know only its support. We can do this by taking each point in the support and finding a polynomial whose support is just that singleton point, then adding all these polynomials together. Finding a polynomial with a given singleton as its support is straightforward — if the point's co-ordinates are $(p_1, \ldots, p_n)$ then the required polynomial is $\prod_{i=1}^{n}(x_i + p_i + 1)$.

So for example suppose $n = 3$ and we wish to find the polynomial whose support is $\{\cdot\cdot 1, 1\cdot 1, 11\cdot, 111\}$ (writing "$\cdot$" rather than "0" for clarity). The required

polynomial is

$$\begin{aligned}&(x_1+1)(x_2+1)x_3 + x_1(x_2+1)x_3 + x_1x_2(x_3+1) + x_1x_2x_3\\ =&x_1x_2x_3 + x_1x_3 + x_2x_3 + x_3 + x_1x_2x_3 + x_1x_3 + x_1x_2x_3 + x_1x_2 + x_1x_2x_3\\ =&x_1x_2 + x_2x_3 + x_3.\end{aligned}$$

An important constraint on bent functions is provided by the following result of Rothaus [35]:

Proposition 1.14: *The degree of a bent function $f : V(2m, 2) \to \mathbb{F}_2$ is at most m unless $m = 1$.*

Proof: Assume $m \geqslant 2$ and pick any monomial $x_{i_1} \dots x_{i_d}$ of degree $d > m$. Let R be the subspace $\langle e_{i_1}, \dots, e_{i_d} \rangle$ of V spanned by the corresponding basis vectors, and write $V = R \oplus S$. Define $g : R \to \mathbb{F}_2$ by $g(r) = f(r \mid 0)$.

Now $|g|$ is odd iff the monomial $x_{i_1} \dots x_{i_d}$ is in f, since in summing over R to evaluate the parity of $|g|$ every other term of f contributes either evenly often or not at all. Thus it is enough to show that $|g|$ is even.

So let $\psi = \omega^g$ and $\varphi = \omega^f$. Considering Fourier transforms on R and V respectively we have

$$\psi(r) = 2^{-d/2} \sum_{t \in R} \tilde{\psi}(t) \omega^{r.t} \qquad \text{and} \qquad \varphi(r \mid 0) = 2^{-n/2} \sum_{y \in V} \tilde{\varphi}(y) \omega^{(r|0).y}.$$

But $\psi(r) = \varphi(r \mid 0)$, so $\psi(\bullet)$ and $\varphi(\bullet \mid 0)$ must have the same Fourier transforms as functions on R. Thus if we write $y = t \mid s$, for all $t \in R$ we have $(r \mid 0).y = r.t$ and hence

$$\tilde{\psi}(t) = 2^{(d-n)/2} \sum_{s \in S} \tilde{\varphi}(t \mid s).$$

Now applying Corollary 1.9 with $x = 0$ and setting $e = \dim S = n - d$ we have

$$\begin{aligned} 2^d - 2|g| = 2^{d/2}\tilde{\psi}(0) &= 2^{d/2}2^{-e/2} \sum_{s \in S} \tilde{\varphi}(0 \mid s) \\ \implies 2^{d-1} - |g| &= 2^{(d-e-2)/2} \sum_{s \in S} \tilde{\varphi}(0 \mid s). \end{aligned}$$

Consider the RHS. Since f is bent each $\tilde{\varphi}(0 \mid s)$ is ± 1. If $d = n$ then $2^{(d-e-2)/2} = 2^{m-1}$ is even and the sum is an integer so the RHS is even. Otherwise the sum involves evenly many ± 1s, so is even, and $d - e$ is at least 2 so $2^{(d-e-2)/2}$ is an integer, so again the RHS is even. Thus in either case since 2^{d-1} is also even we see that $|g|$ must be even. $\square$

If $m = 1$ (so $n = 2$) then the bent functions are precisely those of degree 2, i.e. the quadratics, as we shall see later.

Proposition 1.14 is one of the main reasons for considering f as a polynomial (rather than just a subset of V) at all — it says that the polynomial representation is likely to be simpler than just listing the points of the support.

Examples of bent functions

The simplest family of examples of bent functions is the set of non-singular quadratic functions (i.e. those of full rank — see Chapter 3). For example consider

$$f : V(4,2) \to \mathbb{F}_2$$
$$(x_1, \ldots, x_4) \mapsto x_1x_2 + x_3x_4.$$

Then (the support of) f is

$$\{11\cdot\cdot, 111\cdot, 11\cdot1, \cdot\cdot11, 1\cdot11, \cdot111\}.$$

Now consider a function in RM_1, for example $\cdots1^{\bar{\perp}}$. Its support is

$$\{\cdots1, 1\cdot\cdot1, \cdot1\cdot1, 11\cdot1, \cdot\cdot11, 1\cdot11, \cdot111, 1111\}.$$

Thus $|f\cap\cdots1^{\bar{\perp}}| = |\{11\cdot1, \cdot\cdot11, 1\cdot11, \cdot111\}| = 2^{n-2}$ as required by Proposition 1.13. Similar checking for the rest of RM_1 shows that f is indeed bent.

Similarly for any even n the function

$$f : V(n,2) \to \mathbb{F}_2$$
$$(x_1, \ldots, x_n) \mapsto x_1x_2 + x_3x_4 + \ldots + x_{n-1}x_n.$$

is bent — we will prove this in Chapter 6.

We can rewrite the condition that f is bent in a number of other forms which are easier to check in various circumstances.

Lemma 1.15: *If $x \neq 0$ then*

(i) $|f + x^{\bar{\perp}}| = |f \cap x^{\perp}| - |f \cap x^{\bar{\perp}}| + 2^{n-1}$.

(ii) $|f + x^{\bar{\perp}}| = |f| - 2|f \cap x^{\bar{\perp}}| + 2^{n-1}$.

Proof:

(i) Points in $f + x^{\bar{\perp}}$ are either in f but not in $x^{\bar{\perp}}$, or in $x^{\bar{\perp}}$ but not in f. The first term on the RHS counts points of the first type, while the other two terms count points of the second type — note that $|x^{\bar{\perp}}| = 2^{n-1}$.

(ii) follows from (i) if we observe that $f = (f \cap x^{\perp}) \sqcup (f \cap x^{\bar{\perp}})$. □

Proposition 1.16: *If $f : V(n,2) \to \mathbb{F}_2$ and $n = 2m$ then the following are equivalent:*

(i) *f is a bent function.*

(ii) $\forall x \in V \quad |f + x^{\bar{\perp}}| = 2^{n-1} \pm 2^{m-1}$.

(iii) $\forall l^? \in \mathrm{RM}_1 \quad |f + l^?| = 2^{n-1} \pm 2^{m-1}$.

(iv) $|f| = 2^{n-1} \pm 2^{m-1}$ *and for all non-zero x in V we have*

$$|f \cap x^{\bar{\perp}}| = \begin{cases} 2^{n-2} - 2^{m-1} \text{ or } 2^{n-2} & \text{if } |f| \text{ is light} \\ 2^{n-2} + 2^{m-1} \text{ or } 2^{n-2} & \text{if } |f| \text{ is heavy.} \end{cases}$$

(v) $\forall x \in V \quad |f \cap x^{\perp}| - |f \cap x^{\bar{\perp}}| = \pm 2^{m-1}$.

Proof:

(i) $\iff$ (ii) is just Proposition 1.10.

(ii) $\iff$ (iii) is clear, since every function in RM_1 is of the form $x^{\bar{\perp}}$ or $x^{\perp}$, and

$$|f + x^{\bar{\perp}}| = 2^{n-1} \pm 2^{m-1} \iff |f + x^{\perp}| = 2^n - (2^{n-1} \pm 2^{m-1}) = 2^{n-1} \mp 2^{m-1}.$$

(iii) $\iff$ (iv): From either (iii) with $l^? = \mathbb{O}$ or from (iv) we know that $|f| = 2^{n-1} \pm 2^{m-1}$, and the result then follows directly from Proposition 1.13.

(ii) $\iff$ (v) is straightforward from Lemma 1.15(i). □

An alternative characterisation

We define the notion of a translate of a set of points in the obvious way:

Definition: If $A \subseteq V$ and v is some vector in V then the *translate of A through v*, A_v is defined by $A_v = \{w + v : w \in A\}$. □

Thus we can write f_v to mean (the support of) f translated through v. For a function in RM_1 we have the following:

Lemma 1.17:

(i) $(x^{\perp})_v = x^{(x.v)}$.

(ii) $(x^{(\sigma)})_v = x^{(\sigma \,\hat{+}\, x.v)}$.

Proof:

(i) $y \in (x^{\perp})_v \iff y + v \in x^{\perp} \iff (y + v).x = 0 \iff y.x = x.v$.

(ii) is immediate from (i). □

We extend this notation as follows:

Definition: If $A, S \subseteq V$ then the *translate of A through S*, A_S is defined by

$$A_S = \bigtriangleup_{v \in S} A_v.$$

□

Thus writing A_v is just a slight abuse of the more general notation $A_{\{v\}}$. We will often consider sets such as $A_{\langle v\rangle}$, which is $A \triangle A_v$ if $v \neq 0$. Note that if v, w are distinct and non-zero then

$$(A_{\langle v\rangle})_{\langle w\rangle} = (A \triangle A_v)_{\langle w\rangle} = A \triangle A_v \triangle A_w \triangle A_{v+w} = A_{\langle v,w\rangle},$$

and so on for larger sets of linearly independent vectors.

It turns out that f is bent precisely when every function $f_{\langle v\rangle}$ (for $v \neq 0$) contains exactly half the points of V. To prove this we introduce the function $\kappa : V(n,2) \to \mathbb{Z}$, defined by:

$$\kappa(v) = 2^{-m}(2^n - 2|f + f_v|).$$

Note that this definition involves $f+f_v$ rather than $f_{\langle v\rangle}$ — this difference is important in the case $v = 0$. Also, while we could cancel some powers of 2 in this definition it will be clearer later to use κ in the form given.

Proposition 1.18: *If $\varphi = \omega^f$ then $\tilde{\kappa} = \tilde{\varphi}^2$.*

Proof: By Lemma 1.8 we have $2^m\kappa(v) = \sum_{x\in V} \omega^{(f+f_v)(x)}$.

To evaluate this, recall that by Proposition 1.5

$$\begin{aligned}
\omega^{f(x)} &= 2^{-m} \sum_{y\in V} \tilde{\varphi}(y)\omega^{y.x} \\
\implies \omega^{f_v(x)} = \omega^{f(x+v)} &= 2^{-m} \sum_{y\in V} \tilde{\varphi}(y)\omega^{y.(x+v)} \\
\implies \omega^{(f+f_v)(x)} &= \omega^{f(x)}\omega^{f_v(x)} \\
&= \left(2^{-m} \sum_{y\in V} \tilde{\varphi}(y)\omega^{y.x}\right)\left(2^{-m} \sum_{z\in V} \tilde{\varphi}(z)\omega^{z.(x+v)}\right) \\
&= 2^{-2m} \sum_{y,z\in V} \tilde{\varphi}(y)\tilde{\varphi}(z)\omega^{y.x \,\hat{+}\, z.x \,\hat{+}\, z.v}
\end{aligned}$$

and hence

$$\kappa(v) = 2^{-m} \sum_{x\in V} \omega^{(f+f_v)(x)} = 2^{-3m} \sum_{x,y,z\in V} \tilde{\varphi}(y)\tilde{\varphi}(z)\omega^{y.x \,\hat{+}\, z.x \,\hat{+}\, z.v}.$$

Now consider the summation over x. The only expression dependent on x is $\omega^{y.x \,\hat{+}\, z.x}$ so by Corollary 1.4 this sum will be 0 unless $z = y$, in which case it will be 2^n. Thus in fact

$$\kappa(v) = 2^{-m} \sum_{y\in V} (\tilde{\varphi}(y))^2 \omega^{y.v}$$

so $\tilde{\varphi}^2$ is the Fourier transform of κ as claimed. □

So now we can prove our alternative characterisation of bent functions — see Preneel *et al.* [34] for more details of this approach:

Theorem 1.19: *A function $f : V(n,2) \to \mathbb{F}_2$ is bent iff $|f_{\langle v \rangle}| = 2^{n-1}$ for all non-zero $v \in V$.*

Proof: Let $\varphi = \omega^f$.

$\Rightarrow$: Since f is bent all the Fourier coefficients $\tilde{\varphi}(x)$ are ± 1. Hence by Proposition 1.18 all the Fourier coefficients $\tilde{\kappa}(x)$ are 1. Taking the Fourier transform of $\tilde{\kappa}$ yields

$$\kappa(v) = 2^{-m} \sum_{x \in X} \tilde{\kappa}(x) \omega^{x.v} = 2^{-m} \sum_{x \in X} \omega^{x.v}.$$

By Lemma 1.3 this sum is 0 if $v \neq 0$ and so

$$2^{-m}(2^n - 2|f + f_v|) = \kappa(v) = 0 \implies |f + f_v| = 2^{n-1}.$$

$\Leftarrow$: By the hypothesis and the definition of κ

$$\kappa(v) = \begin{cases} 0 & \text{if } v \neq 0 \\ 2^m & \text{if } v = 0. \end{cases}$$

Hence using Proposition 1.18 we have that for all $x \in V$

$$(\tilde{\varphi}(x))^2 = \tilde{\kappa}(x) = 2^{-m} \sum_{v \in X} \kappa(v) \omega^{v.x} = 2^{-m} 2^m = 1 \qquad \implies \qquad \tilde{\varphi}(x) = \pm 1$$

and hence f is bent. □

In Chapter 6 we will use this characterisation to prove that a non-singular quadratic function is bent.

2. Motivation from coding theory

Reed-Muller codes

As we saw in Chapter 1 given a vector space V of dimension n over $K = \mathbb{F}_q$ the functions from V to K of degree at most r form a vector (sub)space $\mathrm{RM}_r(n)$. We can regard this as a linear code, the rth order Reed-Muller code (see Hill [12], for example). The positions of the code correspond to the points of V. The words of the code correspond to the functions in RM_r — a word just consists of the corresponding function evaluated at each point of V. The (Hamming) distance between two words is the number of positions in which they differ, as in Chapter 1.

We write the parameters of a linear code as [word length, dimension, minimum distance]. From now on we will mainly consider binary codes, i.e. those with $q = 2$.

Proposition 2.1: $\mathrm{RM}_r(n)$, *the space of functions on* $V(n, 2)$ *of degree at most* r, *is a binary linear code with parameters* $\left[2^n, \sum_{i=0}^{r} \binom{n}{i}, 2^{n-r}\right]$.

Proof: The code is clearly binary linear, since addition of codewords corresponds to addition of functions, and adding two functions of degree at most r produces a function of degree at most r. It is also clear that the word length is $|V| = 2^n$.

There is a 1–1 correspondence between functions and their polynomial representations, and the polynomials of degree at most r have the monomials of degree at most r as a basis. The dimension of a Reed-Muller code is therefore just the number of these monomials. If $K = \mathbb{F}_2$ this is just $\sum_{i=0}^{r} \binom{n}{i}$, since a monomial of degree i is a product of exactly i of the n possible variables.

The minimum distance is proved by Cameron and van Lint [7 Theorem 12.2] using induction on n as follows:

If $r = n$ then $\mathrm{RM}_r(n)$ is the whole space V, and clearly has minimum distance 1. If $r < n$ pick a word $w \neq 0$ of the code, and write it as $w = u' + v'$ where u' (respectively v') is a sum of monomials not involving (involving) x_n. Then with respect to an obvious ordering of the code's positions we can write $u' = (u, u)$ for some word $u \in \mathrm{RM}_r(n-1)$, and $v' = (0, v)$ for some word $v \in \mathrm{RM}_{r-1}(n-1)$.

Now if $v \neq u$ then u and $u+v$ are both in $\mathrm{RM}_r(n-1)$ and so $w = (u, u+v)$ has weight at least $2.2^{(n-1)-r} = 2^{n-r}$. Otherwise $u \in \mathrm{RM}_{r-1}(n-1)$ and so $w = (u, 0)$ has weight at least $2^{(n-1)-(r-1)} = 2^{n-r}$. □

The code induced by a bent function

We can consider any subset of the positions of a linear code to obtain a second linear code, called a *punctured subcode* of the original one. So, given a bent function f on $V(n,2)$ we can form a punctured subcode of the 1st-order binary Reed-Muller code RM_1 by considering those positions in the support of f. We denote the resultant code by $\mathcal{C}(f)$. If $n=2$ then $\mathcal{C}(f)$ contains every possible word, so is not very useful. However, larger cases are more interesting:

Proposition 2.2: *If f is light and $n>2$ then $\mathcal{C}(f)$ is a binary linear code with parameters $[2^{n-1}-2^{m-1}, n+1, 2^{n-2}-2^{m-1}]$.*

Proof: $\mathcal{C}(f)$ inherits the binary linearity from RM_1, and clearly has length $|f|$.

A word w of $\mathcal{C}(f)$ is obtained by evaluating the corresponding linear function $l^?$, say, on each of the points of $\mathrm{supp}(f)$. Thus the weight of w is the number of points in the support of f which are also in the support of $l^?$, i.e. $\mathrm{wt}(w)=|f\cap l^?|$. But for $l^?\neq\mathbb{O}$ this RHS is at least the claimed minimum weight, by Proposition 1.13.

This last observation also shows that only $l^?=\mathbb{O}$ can give the zero word, so $\mathcal{C}(f)$ must have the same dimension as RM_1, i.e. $n+1$ by Proposition 2.1. □

If instead we take f to be a heavy bent function we obtain a binary linear code with parameters $[2^{n-1}+2^{m-1}, n+1, 2^{n-2}]$. However (for $n>2$) this code does not seem as good as the light version, since we have increased the word length by 2^m while only increasing the minimum distance by 2^{m-1}. Thus we will usually consider only codes arising from light bent functions, although most of the results can be adapted to the heavy case.

Another way to regard this construction is as follows — given a light bent function we write the points in its support as column vectors of co-ordinates to produce a matrix. We then use the rows of this matrix, together with $\mathbb{1}$, the all-1 row, as generators for the linear code.

The $n=4$ case is an easy example. We noted in Chapter 1 that the bent function $f=x_1x_2+x_3x_4$ has support $\{11\cdot\cdot, 111\cdot, 11\cdot1, \cdot\cdot11, 1\cdot11, \cdot111\}$. If we consider a generating matrix for RM_1 and find the appropriate punctured subcode (indicated by ↓s above the matrix) we see that its columns are just the co-ordinates of the support followed by a 1:

$$
\begin{array}{cc}
 & \downarrow\ \ \downarrow\ \ \downarrow\downarrow\downarrow\downarrow \\
e_1^{\perp} & \cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1 \\
e_2^{\perp} & \cdot\cdot11\cdot\cdot11\cdot\cdot11\cdot\cdot11 \\
e_3^{\perp} & \cdot\cdot\cdot\cdot1111\cdot\cdot\cdot\cdot1111 \\
e_4^{\perp} & \cdot\cdot\cdot\cdot\cdot\cdot\cdot\cdot11111111 \\
\mathbb{1} & 1111111111111111
\end{array}
\quad \xrightarrow{\text{puncture}} \quad
\begin{array}{c}
111.1. \\
111..1 \\
.1.111 \\
..1111 \\
111111
\end{array}
$$

By Proposition 2.2 the code has parameters $[6, 5, 2]$ and we can see from the matrix that it is just the even-weight binary code of length 6.

Weight enumerators

The (Hamming) weights of the various words of a code, i.e. the numbers of non-zero symbols, can be encoded in a polynomial called the *weight enumerator* of the code:

Definition: The *weight enumerator* of a code $\mathcal{C}$ is $W_{\mathcal{C}}(z) = \sum_{c \in \mathcal{C}} z^{\mathrm{wt}(c)}$. □

A number of useful facts about a code, for example its minimum weight, can be read directly from its weight enumerator. It is easy to find the weight enumerator of $\mathcal{C}(f)$:

Proposition 2.3: *If f is a light bent function of $n = 2m$ variables then*

$$W_{\mathcal{C}(f)}(z) = z^{2^{n-1}-2^{m-1}} + (2^n - 1)z^{2^{n-2}} + (2^n - 1)z^{2^{n-2}-2^{m-1}} + 1.$$

Proof: As we noted in the proof of Proposition 2.2, Proposition 1.13 shows that only words of the claimed weights occur. The all-0 and all-1 words are in $\mathcal{C}$ and each is the only word of its weight, so we have accounted for the outer two terms of $W_{\mathcal{C}}$. Furthermore we know that the remaining two coefficients must sum to $2^{n+1} - 2$ since $\mathcal{C}$'s dimension is $n + 1$.

Now, since $\mathcal{C}$ contains the all-1 word, there is a 1–1 correspondence between words in $\mathcal{C}$ of weight 2^{n-2} and their complements, which have weight $2^{n-2} - 2^{m-1}$. Thus the coefficients of the middle two terms of $W_{\mathcal{C}}$ must be equal, and must therefore have the claimed values. □

Thus we can see that there are very few weights possible for words in $\mathcal{C}(f)$. This can be useful for decoding, since when we receive a word we can often quickly find the number of errors which have occurred (or rather the number we must assume have occurred). For example, if $n = 6$ then the code has parameters $[28, 7, 12]$ and contains words of weights 0, 12, 16 and 28 only. If we receive a word of weight 19 then, assuming an original word of these four weights, the smallest number of errors which can have occurred is 19, 7, 3, 9 respectively. Thus since the code can detect up to 6 errors we must assume that the original word had weight 16 and that 3 errors have occurred.

In fact all codes with parameters and weight enumerator of this form arise from bent functions in this way:

Theorem 2.4: *If $\mathcal{C}$ is a binary linear code with parameters*

$$[2^{n-1} - 2^{m-1}, n + 1, 2^{n-2} - 2^{m-1}]$$

and weight enumerator

$$W_{\mathcal{C}(f)}(z) = z^{2^{n-1}-2^{m-1}} + (2^n - 1)z^{2^{n-2}} + (2^n - 1)z^{2^{n-2}-2^{m-1}} + 1$$

then (up to permutation of its positions) it arises as $\mathcal{C}(f)$ for some bent function f on $V(n, 2)$.

Proof: Write down a generating matrix G for the code which contains the all-1 row — start with the all-1 row and then repeatedly pick words not in the span of the rows picked so far. Delete the all-1 row to get G' and then label the ith row with e_i. Now interpret each column as a vector of co-ordinates specifying a point in $V(n, 2)$. Let f denote this set of points.

Pick a point $0 \neq x \in V(n, 2)$. It can be written as a sum of the basis vectors $e_1, \ldots, e_n$ — let w be the sum of the corresponding rows of G'. Pick a position p of the code and let v be the point of f specified by column p of G'. Now w has a 1 at p iff oddly many of the corresponding rows of G' have a 1 at p iff $x.v = 1$. Thus summing over p we see that $\mathrm{wt}(w)$ is just $|f \cap x^{\bar{\perp}}|$.

Now $\mathrm{wt}(w)$ cannot be 0 or 2^n since $x \neq 0$ and the rows of G are linearly independent, so by the weight enumerator assumption $\mathrm{wt}(w) = 2^{n-2}$ or $2^{n-2} - 2^{m-1}$. Since x was arbitrary Proposition 1.16 tells us that f is bent. □

In fact if we want a code with this length and dimension which contains the all-1 vector and has words of only four weights then this minimum distance is the best possible. This follows from the first two paragraphs of the proof of Theorem 2.4, which apply to any such code, together with the fact that balls of radius $2^{n-1} - 2^{m-1}$ around the codewords of RM_1 cover the whole of $V(n, 2)$ — see Preneel *et al.* [34 Theorem 3.4], for example.

Although different bent functions induce apparently different punctured subcodes of the 1st-order Reed-Muller code, in many cases these codes are in fact essentially the same.

Definition: We say that two codes are *equivalent* if one can be obtained from the other by permuting its columns and multiplying each column by a non-zero element of the underlying field K. □

These operations apply to the complete set of words in the code. Thus, if we want to know that the linear codes generated by two given matrices are equivalent we may have to apply a change of basis to the matrix also, i.e. pre-multiply it by a non-singular square matrix. In Chapter 3 we will see the extent to which these equivalences correspond to equivalences of the original bent functions.

Other applications to coding theory and cryptography

Summaries of various desirable properties for cryptographic functions are provided by Meier and Staffelbach [30], Nyberg [31] and Preneel *et al.* [34]. These properties essentially require that a function f is "nonlinear" or "random", in various senses. For example the degree of the polynomial representation of f (the *nonlinear order*) should be high, and it should be 0/1 balanced. Another such property is that of being *perfect nonlinear (with respect to linear structures)*, which is exactly the property required by Theorem 1.19 for a function to be bent.

The point of this property is that it implies that the function has the largest possible distance to the set of linear functions — this makes it cryptographically strong since many cryptanalytic methods rely on finding linearities in the system being broken. Such a function is "as far from linear as possible", hence the name "bent". Note that a bent function also has the largest possible distance to the set of affine (RM_1) functions, which can be cryptographically important.

This property can also be interpreted in terms of the functions' autocorrelations, (the correlations between the function and its translates). These are important in the construction of stream ciphers [30] and bent sequences [32]. The best possible autocorrelation properties are provided by bent functions, although in practice the function may be modified slightly to improve performance with respect to other criteria, for example nonlinear order.

3. Equivalent bent functions

If f is a bent function on $V(n,2)$ there are a number of simple operations which we can apply to V to obtain new bent functions, as described by Rothaus [35].

Definition: A *linear automorphism* of a vector space V over K is a bijective map α from V to V satisfying

$$\alpha(\lambda x + \mu y) = \lambda\alpha(x) + \mu\alpha(y) \quad \forall \lambda, \mu \in K \text{ and } x, y \in V.$$

Note that if $K = \mathbb{F}_2$ we need consider only $\alpha(x+y)$, i.e. the case $\lambda = \mu = 1$. □

Any such map can be represented as a matrix acting on the points written as column vectors of co-ordinates with respect to some basis. The set of such maps forms a group under composition, the *general linear group* $\mathrm{GL}(V)$, or $\mathrm{GL}_n(q)$ if $K = \mathbb{F}_q$, which we met in Chapter 1.

We need to know how these maps act on various functions. If $A \subseteq V$ then we write $\alpha A = \{\alpha a : a \in A\}$, so αf is the set of images under α of points in (the support of) f.

Also, since any $\alpha \in \mathrm{GL}(V)$ can be written as a matrix, we can define α^T, the map obtained by transposing this matrix. Since

$$(\alpha^{-1})^T = (\alpha^{-1})^T \mathbb{I} = (\alpha^{-1})^T \alpha^T (\alpha^T)^{-1} = (\alpha\alpha^{-1})^T (\alpha^T)^{-1} = \mathbb{I}^T (\alpha^T)^{-1} = (\alpha^T)^{-1}$$

transposition commutes with taking inverses, so α^{-T} is unambiguous.

We note some simple results about the action of a linear automorphism:

Lemma 3.1: *If $A \in V$, $\alpha \in \mathrm{GL}(V)$, $v \in V$ then*
(i) $\alpha(x^\perp) = (\alpha^{-T}x)^\perp$.
(ii) $\alpha(A_v) = (\alpha A)_{\alpha v}$.

Proof:
(i)

$$\begin{aligned} y \in \alpha(x^\perp) &\iff \alpha^{-1}y \in x^\perp \iff (\alpha^{-1}y)^T x = 0 \iff y^T \alpha^{-T} x = 0 \\ &\iff y \in (\alpha^{-T}x)^\perp. \end{aligned}$$

(ii)

$$\begin{aligned} y \in \alpha(A_v) &\iff \alpha^{-1}y + v \in A \iff \alpha^{-1}(y + \alpha v) \in A \iff y + \alpha v \in \alpha A \\ &\iff y \in (\alpha A)_{\alpha v}. \end{aligned}$$

□

We can now find a large class of bent functions equivalent to our chosen bent function f:

Proposition 3.2: *If $f : V \to \mathbb{F}_2$ is a bent function, $\alpha \in \mathrm{GL}(V)$ and $v \in V$ then*

(i) *αf is a bent function.*

(ii) *$f + v^?$ is a bent function.*

(iii) *f_v is a bent function.*

Proof: Using Proposition 1.16 to check for bentness we have

(i) $|\alpha f + x^?| = |\alpha(f + \alpha^{-1}x^?)| = |f + \alpha^{-1}x^?|$ since α is a bijection. But $x^?$ ranges over RM_1 as $\alpha^{-1}x^? = (\alpha^T x)^?$ does, so checking that f is bent is equivalent to checking that αf is bent.

(ii) $|f+v^?+x^?| = |f+(v+x)^?|$ and $x^?$ ranges over RM_1 as $(v+x)^?$ does, so checking that f is bent is equivalent to checking that $f + v^?$ is bent.

(iii) $|f_v + x^?| = |(f + x^?)_v| = |f + x^?|$, so checking that f is bent is equivalent to checking that f_v is bent. □

Each of these three types of map forms a group acting on the subsets of V. In the light of Lemmas 1.17 and 3.1 the most general form of a composition of actions of the three types on A is the following:

Definition: If $\alpha \in \mathrm{GL}(V)$, $a \in V$ and $b^{(\sigma)} \in \mathrm{RM}_1$ then the map $[\alpha, a, b^{(\sigma)}]$ acts on subsets of V as follows:

$$[\alpha, a, b^{(\sigma)}]A = (\alpha A)_a + b^{(\sigma)}.$$

□

The set of maps $[\alpha, a, b^{(\sigma)}]$ forms a group under composition. Proposition 3.2 says that these maps take bent functions to bent functions so we call them the *general bent-function-preserving* maps, and write $\mathrm{GB}(V)$ for the group.

Definition: We call two bent functions *equivalent* if they are in the same orbit under $\mathrm{GB}(V)$. If a bent function f has some property expressed by "f is P" we say that any bent function equivalent to it is "*essentially P*". □

For example we might say that the bent functions of 4 variables in the same orbit as $x_1x_2 + x_3x_4$ are "essentially $x_1x_2 + x_3x_4$".

We can find the size of $\mathrm{GB}(V)$ easily, since we have the following:

Lemma 3.3: *Two maps $[\alpha, a, b^{(\sigma)}]$ and $[\beta, c, d^{(\tau)}]$ are equal iff the ordered triples $(\alpha, a, b^{(\sigma)})$ and $(\beta, c, d^{(\tau)})$ are equal.*

Proof:

$\Leftarrow$: Trivial.

$\Rightarrow$: Consider the maps' action on a halfspace $x^?$ — by Lemmas 1.17 and 3.1

$$[\alpha, a, b^?]x^? = (\alpha x^?)_a + b^? = \left((\alpha^{-T}x)^? + b^?\right)_a = (\alpha^{-T}x + b)^?.$$

Thus if the actions of $[\alpha, a, b^{(\sigma)}]$ and $[\beta, c, d^{(\tau)}]$ on such pairs are the same then $\alpha^{-T}x + b = \beta^{-T}x + d$ for all $x \in V$. Taking $x = 0$ shows that $b = d$, and then clearly α and β must also be equal. Now by considering the maps' actions on $\{0\}$ it is clear that $a = c$ and $\sigma = \tau$. □

Now to construct a matrix $\alpha \in \mathrm{GL}(V)$ we must pick a non-zero first row, then successively pick rows not in the span of those picked so far. Hence the number of possible α is just

$$|\,\mathrm{GL}(V)| = (2^n - 2^0)(2^n - 2^1)\dots(2^n - 2^{n-1}).$$

The number of choices for a is 2^n, while the number of choices of $b^{(\sigma)}$ is 2^{n+1}. Thus we have

$$|\,\mathrm{GB}(V)| = 2^{2n+1}(2^n - 2^0)(2^n - 2^1)\dots(2^n - 2^{n-1}).$$

We can rewrite $\mathrm{GB}(V)$'s multiplication and inversion operations more directly:

Theorem 3.4: *The elements of* $\mathrm{GB}(V)$ *satisfy*

(i) $[\alpha, a, b^{(\sigma)}][\beta, c, d^{(\tau)}] = [\alpha\beta, a + \alpha c, (b + \alpha^{-T}d)^{(\sigma \,\hat{+}\, \tau \,\hat{+}\, a.\alpha^{-T}d \,\hat{+}\, 1)}]$.

(ii) $[\alpha, a, b^{(\sigma)}]^{-1} = [\alpha^{-1}, \alpha^{-1}a, (\alpha^T b)^{(\sigma \,\hat{+}\, a.b)}]$.

Proof: Using Lemmas 1.17 and 3.1 we have

(i)

$$\begin{aligned}
[\alpha, a, b^\sigma][\beta, c, d^\tau]f &= (\alpha((\beta f)_c + d^\tau))_a + b^\sigma \\
&= ((\alpha\beta f)_{\alpha c} + \alpha d^\tau)_a + b^\sigma \\
&= (\alpha\beta f)_{a+\alpha c} + (\alpha^{-T}d)^{(\tau \,\hat{+}\, a.\alpha^{-T}d)} + b^\sigma \\
&= [\alpha\beta, a + \alpha c, (b + \alpha^{-T}d)^{(\sigma \,\hat{+}\, \tau \,\hat{+}\, a.\alpha^{-T}d \,\hat{+}\, 1)}]f.
\end{aligned}$$

(ii) This is easily checked using (i). □

Equivalences of codes induced by bent functions

There is a close connection between equivalences of bent functions and equivalences of their induced codes — recall that the induced code $\mathcal{C}(f)$ is the punctured subcode of RM_1 whose positions are the points in the support of f. As we saw in Chapter 2 we obtain a generating matrix M for $\mathcal{C}(f)$ by writing down the co-ordinates of the points in the support of f with respect to some basis $\{e_1, \ldots, e_n\}$ of V as column vectors and appending the all-1 row. Let N be the co-ordinate portion of the matrix, i.e. all but the bottom all-1 row.

Now consider applying a linear automorphism α to V to obtain the code $\mathcal{C}(\alpha f)$ with generating matrix M'. Given a point $x \in \mathrm{supp}(f)$ we obtain a point $y \in \mathrm{supp}(\alpha f)$ by applying α to x. But if α corresponds to a matrix A with respect to $\{e_1, \ldots, e_n\}$ then its action on x written as a column vector of co-ordinates is just left multiplication: $x \mapsto Ax$. Thus we see that

$$N' = AN \qquad \Longrightarrow \qquad M' = \left(\begin{array}{c|c} A & \mathbb{O} \\ \hline \mathbb{O} & 1 \end{array}\right) M$$

and so $\mathcal{C}(f)$ and $\mathcal{C}(\alpha f)$ are equivalent.

Translating f through a vector v is similar. Given a point $x \in \mathrm{supp}(f)$ we obtain the column vector representing a point $y \in \mathrm{supp}(f_v)$ by adding the column vectors representing v and x. Thus to get N' we must add the all-1 row to those rows of N corresponding to the 1s of v. In other words

$$M' = \left(\begin{array}{c|c} \mathbb{I}_n & v \\ \hline \mathbb{O} & 1 \end{array}\right) \left(\begin{array}{c} N \\ \hline \mathbb{1} \end{array}\right) = \left(\begin{array}{c|c} \mathbb{I}_n & v \\ \hline \mathbb{O} & 1 \end{array}\right) M$$

and so as before $\mathcal{C}(f)$ and $\mathcal{C}(f_v)$ are equivalent.

Adding an RM_1 function to f, on the other hand, can produce *inequivalent* codes — there are examples in the $n = 8$ case.

Equivalence classes of bent functions

With our notion of equivalent bent functions, we can ask a number of obvious questions about the equivalence classes for each even n. How many are there? How large are they? For $n \leqslant 6$ these questions are reasonably easy to answer, but first we need to find out more about equivalent functions.

If we consider a linear automorphism α of V as acting on sums of the basis vectors $\{e_1, \ldots, e_n\}$ then we can define a corresponding action on sums of the variables $\{x_1, \ldots, x_n\}$. We can then extend this to an action on polynomials in the variables by saying that the image of a sum (respectively product) of variables is the sum (product) of their images. We write this action as a superscript.

Thus for example if $n = 2$, α is represented by the matrix $\begin{smallmatrix}1&1\\0&1\end{smallmatrix}$ and p is the polynomial $x_1x_2 + x_2$ then α sends $x_1 \mapsto x_1 + x_2$ and $x_2 \mapsto x_2$, so

$$p^\alpha = (x_1 + x_2)x_2 + x_2 = x_1x_2 + x_2 + x_2 = x_1x_2.$$

This action is closely related to the direct action of α on the support of p:

Proposition 3.5: $\alpha(\mathrm{supp}(p)) = \mathrm{supp}\left(p^{\alpha^{-T}}\right)$.

Proof: First consider the case in which p is a single variable. Without loss of generality $p = x_1$ so $\mathrm{supp}(p) = {e_1}^{\bar{\perp}}$. Then the result follows directly from Lemma 3.1:

$$x \in \mathrm{LHS} \iff x \in \alpha({e_1}^{\bar{\perp}}) \iff x \in (\alpha^{-T}e_1)^{\bar{\perp}}$$

which is the support of the polynomial ${e_1}^{\alpha^{-T}}$.

Now if p is a monomial $\prod_{i\in I} x_i$ then $\mathrm{supp}(p) = \cap_{i\in I} {e_i}^{\bar{\perp}}$ and so

$$x \in \mathrm{LHS} \iff x \in \bigcap_{i\in I} \alpha({e_i}^{\bar{\perp}}) \iff x \in \bigcap_{i\in I} (\alpha^{-T}e_i)^{\bar{\perp}}$$

which is the support of $\prod_{i\in I} {e_i}^{\alpha^{-T}}$.

Finally if p is a sum of monomials $\sum_{i\in I} m_i$ then $\mathrm{supp}(p) = \triangle_{i\in I}\,\mathrm{supp}(m_i)$ and so

$$x \in \mathrm{LHS} \iff x \in \bigtriangleup_{i\in I} \alpha(\mathrm{supp}(m_i)) \iff x \in \bigtriangleup_{i\in I} \mathrm{supp}\left({m_i}^{\alpha^{-T}}\right)$$

which is the support of $\sum_{i\in I} {m_i}^{\alpha^{-T}}$. □

We can consider translations in terms of a similar action on polynomials. To translate through a vector v we write v as a sum of basis vectors and then replace x_i by $x_i + 1$ in the polynomial representation of f for those x_is corresponding to basis vectors in this sum.

Thus for example if $n = 2$, $v = e_2$ and p is the polynomial $x_1x_2 + x_2$ then translation by v sends $x_1 \mapsto x_1$ and $x_2 \mapsto x_2 + 1$, so

$$p_v = x_1(x_2 + 1) + (x_2 + 1) = x_1x_2 + x_1 + x_2 + 1.$$

Thus we can now calculate the action of a map $[\alpha, a, b^{(\sigma)}]$ on (the polynomial representation of) a function f directly, rather than having to calculate the image of $\mathrm{supp}(f)$ and then find the polynomial corresponding to this image. This is important because as we saw in Chapter 1 it is usually easier to deal with the polynomial representation than directly with the support of f.

Quadratic forms

A particularly important case of equivalence is that of quadratic forms on V — see Dickson [10], for example. To start with we consider homogeneous quadratic forms, since we can adjust the lower-degree terms by the addition of a function in RM_1.

Any homogeneous quadratic function f on V can be identified with a symmetric $n \times n$ matrix over $\mathbb{F}_2$ with empty diagonal (i.e. such that every entry on the main diagonal is 0) as follows — we write

$$f(x) = \sum_{1 \leqslant i < j \leqslant n} a_{ij} x_i x_j$$

where each a_{ij} is 1 or 0 according as $x_i x_j$ is or is not a term of the quadratic — recall that f cannot have terms of the form ${x_i}^2$. Then we consider the a_{ij}s as entries of a matrix A, and make this a symmetric matrix (with empty diagonal) by setting $a_{ij} = a_{ji}$ for $i > j$. This symmetric matrix has a rank, and we call this the rank of the homogeneous quadratic form (and those formed by adding RM_1 functions to it). Hence we say that a quadratic form is *non-singular* if the associated symmetric matrix has full rank.

Now consider applying a linear automorphism to V and hence to the support of f. By Proposition 3.5 this is equivalent to applying a linear automorphism to the argument x. If the automorphism is represented by a matrix B (with respect to some basis of V) it sends $x_i \mapsto \sum_k b_{ik} x_k$. Hence

$$f(Bx) = \sum_{i<j} \left(a_{ij} \sum_k b_{ik} x_k \sum_l b_{jl} x_l \right) = \sum_{i<j} \sum_{k,l} a_{ij} b_{ik} x_k b_{jl} x_l.$$

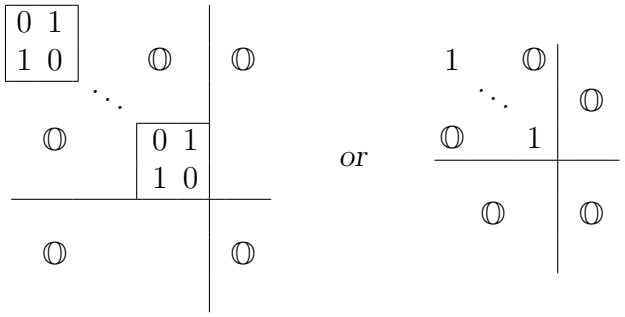
But the summand $a_{ij} b_{ik} x_k b_{jl} x_l$ is unchanged if we interchange the pairs $\{i, j\}$, $\{k, l\}$. Thus instead of summing over $i < j$ and all k, l, we can equivalently sum over all i, j and $k < l$. Then

$$f(Bx) = \sum_{k<l} \left(\sum_{i,j} b_{ik} a_{ij} b_{jl} \right) x_k x_l = \sum_{k<l} (B^T A B)_{kl} x_k x_l.$$

Thus we have changed the quadratic form represented by A to the one represented by $B^T A B$ — as we shall see this is also symmetric with empty diagonal. Since B is non-singular $B^T A B$ has the same rank as A, so their associated quadratic forms also have the same rank.

So now we can prove the following, which in fact proves rather more than we need, for use in later chapters:

Proposition 3.6: *If A is a symmetric matrix over $\mathbb{F}_2$ and $B \in \mathrm{GL}(V)$ then $B^T AB$ is symmetric, has the same rank as A and has empty diagonal iff A does. Moreover there exists some $B \in \mathrm{GL}(V)$ such that $B^T AB$ has one of these two forms:*



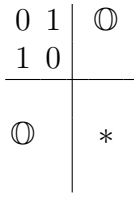
according as A does or does not have empty diagonal.

Proof: If B is a permutation matrix then $A \mapsto B^T AB$ consists of applying this permutation to the rows and columns of A. Similarly if $B = \mathbb{I}_n + \mathbb{E}_{ij}$, where $\mathbb{E}_{ij}$ is the matrix with exactly one non-zero entry at (i, j), then $A \mapsto B^T AB$ consists of adding the ith row and column of A to the jth row and column respectively.

Operations of both these types preserve A's symmetry and whether or not it has empty diagonal — the permutations just permute the elements on the main diagonal while the effects of the row and column additions on elements of the main diagonal cancel out. These operations also preserve A's rank since the various Bs are non-singular. Since they generate the whole of $\mathrm{GL}(V)$ (see [10], for example) we have the first sentence of the result.

For the second sentence we use these elementary operations to put A into the standard forms:

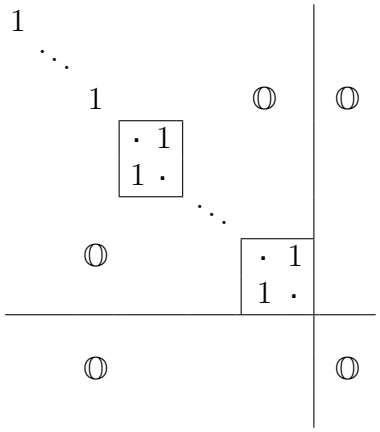
Case 1 A has empty diagonal: Either A is empty and we're done or else we can use permutations to put a 1 in the second column of the first row and hence, by symmetry, another 1 in the first column of the second row. The second row (respectively column) is now $10 * \ldots *$ so we can add it to appropriate rows below it (columns to the right of it) to make the rest of the first column (row) 0. The first row (respectively column) is now $010 \ldots 0$ so we can add it to appropriate rows below it (columns to the right of it) to make the rest of the second column (row) 0. So far we have the matrix



Now we can carry out the same algorithm on the $*$ portion — since we only ever permute and add together rows below the first two and columns to the right of the

first two this can never alter the other portions. Applying the algorithm repeatedly puts A into the first standard form.

Case 2 A has a 1 somewhere on its main diagonal: We can use permutations to put this 1 in A's top left corner. We can add the first row (respectively column) to appropriate rows (columns) below it to make the rest of the first column (row) 0. As before we can either carry out this algorithm on the remaining portion of the matrix, or use Case 1 here. Repeating this process we obtain



We can now convert the $\left(\begin{smallmatrix}\cdot & 1\\ 1 & \cdot\end{smallmatrix}\right)$ blocks into $\left(\begin{smallmatrix}1 & \cdot\\ \cdot & 1\end{smallmatrix}\right)$ blocks by conjugations of the form

$$\begin{pmatrix}111\\1\cdot1\\11\cdot\end{pmatrix}\begin{pmatrix}1\cdot\cdot\\\cdot\cdot1\\\cdot1\cdot\end{pmatrix}\begin{pmatrix}111\\1\cdot1\\11\cdot\end{pmatrix}=\begin{pmatrix}111\\11\cdot\\1\cdot1\end{pmatrix}\begin{pmatrix}111\\1\cdot1\\11\cdot\end{pmatrix}=\begin{pmatrix}1\cdot\cdot\\\cdot1\cdot\\\cdot\cdot1\end{pmatrix}$$

and thus put A into the second standard form. □

Corollary 3.7: *All symmetric $n \times n$ matrices of the same rank with empty (respectively non-empty) diagonal are conjugate in the above sense.*

Proof: They are all conjugate to the appropriate standard form. □

Corollary 3.8: *Any two homogeneous quadratic functions of the same rank on $V(n, 2)$ are equivalent via linear automorphisms.* □

Corollary 3.9: *Any two non-singular quadratic functions on $V(n, 2)$ are equivalent via actions of* $\mathrm{GB}(V)$.

Proof: The degree 2 terms are equivalent via a linear automorphism, and we can adjust the lower-degree terms by adding RM_1 functions. □

Corollary 3.10: *A quadratic function has even rank.*

Proof: The first standard matrix has even rank. □

Corollary 3.11: *If $n = 2m$ then a homogeneous quadratic function f of rank $2r$ on $V(n, 2)$ has weight $2^{n-1} - 2^{n-r-1}$.*

Proof: By Corollary 3.8 f and $g = x_1x_2 + x_3x_4 + \ldots + x_{2r-1}x_{2r}$ are equivalent via a linear automorphism, hence have the same weight. We find the weight of g by induction on r:

If $r = 0$ then $g = \mathbb{O}$ which has weight 0 as claimed. If $r > 0$ then $h = x_1x_2 + \ldots + x_{2r-3}x_{2r-2}$ has weight $2^{n-1} - 2^{n-r}$ by the inductive hypothesis. The support of g consists of $\frac{3}{4}$ of the support of h, namely the points with $x_{2r-1}x_{2r} = 0$, together with $\frac{1}{4}$ of its complement, namely the points with $x_{2r-1}x_{2r} = 1$. Thus

$$\mathrm{wt}(f) = \mathrm{wt}(g) = \tfrac{3}{4}|h| + \tfrac{1}{4}(2^n - |h|) = \tfrac{1}{2}(2^{n-1} - 2^{n-r}) + \tfrac{1}{4}2^n = 2^{n-1} - 2^{n-r-1}$$

as claimed. □

More about equivalence classes of bent functions

Now we are able to answer our questions about equivalence classes of bent functions for all $n \leqslant 6$:

$\underline{n = 2}$ Clearly f cannot be in RM_1 (since if it were then by Proposition 1.16 we would have $2^{n-1} \pm 2^{m-1} = |f + f| = |\mathbb{O}| = 0$). Thus since f has degree at most n it must have degree exactly 2, i.e. it is a quadratic — recall that the degree bound of Proposition 1.14 does not hold for this case. In fact in this case the bent functions are precisely the non-singular quadratics, and equivalently are precisely the functions of odd weight.

By Corollaries 1.11 and 3.11 we know that f must be non-singular. Since all non-singular homogeneous quadratics are equivalent, and we can always adjust the linear + constant part of a function by equivalences, any two bent functions on $V(2, 2)$ are equivalent.

$\underline{n = 4}$ As for $n = 2$, f cannot be in RM_1 and has degree at most $m = 2$ by Proposition 1.14, so is a non-singular quadratic. Thus as before any two bent functions on $V(4, 2)$ are equivalent.

$\underline{n = 6}$ In this case Proposition 1.14 tells us only that $\deg(f) \leqslant 3$. As in the previous cases we know there is a class of non-singular quadratics, but there may be bent functions of degree 3 also.

Unfortunately the cubic functions of 6 variables are not all equivalent. In Proposition 3.6 we considered a matrix corresponding to a quadratic — in the same way we can consider a 3-dimensional array corresponding to a function of degree 3 (a

cubic function), and use elementary *plane* operations to put such an array into a standard form.

We can then consider adding all possible quadratic functions to representatives of each of these classes and check which of the resultant functions is bent. By considering the stabilisers of the cubic parts we can pick representatives for the various orbits of "good" quadratics.

This calculation is described by Rothaus [35] and Parker, Spence and Tonchev [33], and has also been made independently by the present author using hand calculation, computer programs written in C and the computer algebra system GAP [25] (versions 2.4, 3.1 and 3.2). It turns out that there are 6 equivalence classes of cubics. For 4 of these we can add a good quadratic to form a bent function, and in each case there is only one orbit of good quadratics. In other words there are exactly 4 equivalence classes of bent functions of 6 variables, with representatives

6.1: $x_1x_2 + x_3x_4 + x_5x_6$
6.2: $x_1x_2x_3 + x_1x_4 + x_2x_5 + x_3x_6$
6.3: $x_1x_2x_5 + x_1x_3x_6 + x_1x_4 + x_2x_3 + x_2x_5 + x_5x_6$
6.4: $x_1x_2x_3 + x_1x_5x_6 + x_2x_4x_6 + x_1x_3 + x_1x_4 + x_2x_3 + x_2x_4 + x_2x_5 + x_3x_6$.

We can calculate the size of the stabiliser of (a representative of) each of our classes, and hence find the actual class sizes, since we know $|\,\mathrm{GB}(V)|$:

Class	$\lvert\,\mathrm{GB}(V)\rvert$	Stabiliser size	Class size
2.1	192	24	8
4.1	10321920	11520	896
6.1	165140150353920	92897280	1777664
6.2		688128	239984640
6.3		122880	1343913984
6.4		43008	3839754240

We shall study these stabilisers in more detail in Chapter 11.

4. Two bent function constructions

The Maiorana construction

Although in Chapter 3 we were able to find several classes of 6-variable bent function, effectively by exhaustive search, so far the only guaranteed classes we have met are the non-singular quadratics. However, Rothaus' construction I [35] gives a generalisation which produces several new classes:

Proposition 4.1: *If $n = 2m$ let g be any function of the m variables $x_1, \ldots, x_m$. Then the function*

$$f = g + x_1x_{m+1} + x_2x_{m+2} + \ldots + x_mx_n$$

is a bent function of n variables. □

Note that this construction yields the non-singular quadratics when $g = \mathbb{O}$, and class 6.2, for example, also arises in this way. Another way to write this construction is to consider V as a direct product $A \oplus B$ of two spaces of dimension m, so that we can write

$$f(x \mid y) = g(x) + x.y.$$

This version leads to a further generalisation by Maiorana, as described by Kumar, Scholtz and Welch [23], for example. In fact in [23] it is shown that this *Maiorana construction* works for generalised bent functions (see Chapter 15), but we need only the $\mathbb{F}_2$ case:

Proposition 4.2 — The Maiorana construction: *If $n = 2m$ let g be any function on $A = \langle x_1, \ldots, x_m \rangle$ and let π be a permutation of the points of A. Then the function*

$$f(x \mid y) = g(x) + (\pi x).y$$

is a bent function on V.

Proof: Kumar, Scholtz and Welch [23 Theorem 1]:

Any function in RM_1 can be written as a sum $r^? + s^?$ where $r \in A$ and $s \in B$, as in Lemma 1.2. Thus we need to consider $|f + r^? + s^?|$ which is

$$\sum_{a\in A}\sum_{b\in B}(g(a) \;\hat{+}\; (\pi a).b \;\hat{+}\; a.r \;\hat{+}\; b.s) = \sum_{a\in A}\sum_{b\in B}(g(a) \;\hat{+}\; a.r \;\hat{+}\; (\pi a + s).b)\,.$$

(recall that "$\hat{+}$" denotes addition in $\mathbb{F}_2$). As b varies the first two terms are constant and the last takes the value 0 as often as it takes the value 1, except when $\pi a = s$.

Thus setting $t = \pi^{-1}s$ we have

$$\begin{aligned} |f + r^? + s^?| &= \sum_{t \neq a \in A} \tfrac{1}{2}|B| + \sum_{b \in B} (g(t) \ \hat{+} \ t.r) \\ &= (2^m - 1)2^{m-1} + 2^m (g(t) \ \hat{+} \ t.r) \\ &= 2^{n-1} \pm 2^{m-1} \end{aligned}$$

according to the value of $(g(t) \ \hat{+} \ t.r)$. Since r and s were arbitrary, f is a bent function by Proposition 1.10. □

For example, let $n = 4$ and let $A = \langle e_1, e_2 \rangle$, $B = \langle e_3, e_4 \rangle$. Now let $g = x_1 + 1$ and let π be the permutation $(\cdot\cdot, \cdot 1, 11)$. Then we can evaluate $f(x \mid y)$ on each point of $V = A \oplus B$, as follows:

x	y	$\pi(x)$	$\pi(x).y$	$g(x)$	$f(x \mid y)$
$\cdot\cdot$	$\cdot\cdot$	$\cdot 1$	$\cdot$	1	1
$1\cdot$	$\cdot\cdot$	$1\cdot$	$\cdot$	$\cdot$	$\cdot$
$\cdot 1$	$\cdot\cdot$	11	$\cdot$	1	1
11	$\cdot\cdot$	$\cdot\cdot$	$\cdot$	$\cdot$	$\cdot$
$\cdot\cdot$	$1\cdot$	$\cdot 1$	$\cdot$	1	1
$1\cdot$	$1\cdot$	$1\cdot$	1	$\cdot$	1
$\cdot 1$	$1\cdot$	11	1	1	$\cdot$
11	$1\cdot$	$\cdot\cdot$	$\cdot$	$\cdot$	$\cdot$
$\cdot\cdot$	$\cdot 1$	$\cdot 1$	1	1	$\cdot$
$1\cdot$	$\cdot 1$	$1\cdot$	$\cdot$	$\cdot$	$\cdot$
$\cdot 1$	$\cdot 1$	11	1	1	$\cdot$
11	$\cdot 1$	$\cdot\cdot$	$\cdot$	$\cdot$	$\cdot$
$\cdot\cdot$	11	$\cdot 1$	1	1	$\cdot$
$1\cdot$	11	$1\cdot$	1	$\cdot$	1
$\cdot 1$	11	11	$\cdot$	1	1
11	11	$\cdot\cdot$	$\cdot$	$\cdot$	$\cdot$

Using the method of adding polynomials with singleton support, we find that this function has polynomial representation

$$f = x_1x_3 + x_1x_4 + x_2x_3 + x_1 + x_4 + 1.$$

This is clearly a non-singular quadratic, hence is a bent function as expected.

The Maiorana construction gives us a large number of bent functions. There are 2^{2^m} choices of the function g, and $2^m!$ choices of the permutation π, so we can obtain at least $2^{2^m}2^m!$ bent functions (see Nyberg [31], for example), although many of them will probably be equivalent.

Furthermore, by using a different choice of A and B, such as $A = \langle e_1, e_3 \rangle$, $B = \langle e_2, e_4 \rangle$ for the above example, it seems that even though the functions produced with different As and Bs might not necessarily be distinct, we would probably obtain many more bent functions.

However, in Chapter 8 we will see how to derive the polynomial representation more directly from g and π. This will allow us to place a constraint on the bent functions which the Maiorana construction can produce, and we will see that in the $n = 8$ case there are bent functions which cannot be constructed in this way.

Duality

We recall from Chapter 1 that if f is bent then by definition the Fourier coefficients $\tilde{\varphi}(x)$ of ω^f, given by

$$\tilde{\varphi}(x) = 2^{-n/2} \sum_{y \in V} \omega^{f(y) \,\hat{+}\, y.x},$$

take only the values ± 1. But in this case we can write $\tilde{\varphi}(x) = \omega^{g(x)}$ for some function $g : V \to \mathbb{F}_2$, and as Rothaus [35] points out this provides another way to construct new bent functions:

Proposition 4.3: *With the above notation, $g(x)$ is a bent function.*

Proof: Corollary 1.6 tells us that the Fourier transform of $\tilde{\varphi}(x)$ is just $\varphi(y)$. In other words, the Fourier coefficients of $\omega^{g(x)}$ are just $\omega^{f(y)}$. But $f(y) = 0$ or 1, so these coefficients take the values ± 1, so $g(x)$ is bent. □

Definition: The function $g(x)$ is called the *dual* of f, and is written f^*. □

The fact that the Fourier transformation is self-inverse (Corollary 1.6) gives us the following immediately:

Proposition 4.4: $f^{**} = f$. □

In order to explore the duality concept, it will be convenient to rewrite the definition of f^*:

Proposition 4.5: *If f is a bent function then*

$$f^*(x) = \begin{cases} 0 & \text{if } |f + x^{\bar{\perp}}| \text{ is low} \\ 1 & \text{if } |f + x^{\bar{\perp}}| \text{ is high.} \end{cases}$$

Proof: This is immediate from Corollary 1.9. □

Note that $f^*(0)$ tells us whether f itself is light or heavy, since $f + 0^{\bar{\perp}} = f$.

If $n = 2$ then the function $f = x_1x_2$ is its own dual, since $\mathrm{supp}(f) = \{11\}$ and so we have

x	$\mathrm{supp}(x^{\bar{\perp}})$	$\mathrm{supp}(f + x^{\bar{\perp}})$	$\lvert\mathrm{supp}(f + x^{\bar{\perp}})\rvert$	$f^*(x)$
$\cdot\cdot$	$\emptyset$	$\{11\}$	1	0
$1\cdot$	$\{1\cdot, 11\}$	$\{1\cdot\}$	1	0
$\cdot 1$	$\{\cdot 1, 11\}$	$\{\cdot 1\}$	1	0
11	$\{1\cdot, \cdot 1\}$	$\{1\cdot, \cdot 1, 11\}$	3	1

However, $(x_1x_2 + x_1)^* = x_1x_2 + x_2$, for example, and in general $f^* \neq f$. Indeed in many, possibly almost all, cases f and f^* are not even equivalent, as we shall see later.

A number of other reworkings of the definition will be useful:

Corollary 4.6: *If f is a bent function then $\forall x \in V$*

(i) $f^*(x) = 2^{-m}(|f + x^{\bar{\perp}}| - 2^{n-1} + 2^{m-1})$.

(ii) $|f + x^{\bar{\perp}}| = 2^{n-1} - 2^{m-1} + 2^m f^*(x)$.

(iii) $|f \cap x^{\bar{\perp}}| = \begin{cases} 2^{n-2} - 2^{m-1} & \text{if } f^*(0) = 0 \text{ and } f^*(x) = 1 \\ 2^{n-2} & \text{if } f^*(0) = f^*(x) \\ 2^{n-2} + 2^{m-1} & \text{if } f^*(0) = 1 \text{ and } f^*(x) = 0. \end{cases}$

(iv) $|f \cap x^{\bar{\perp}}| - |f \cap x^{\perp}| = 2^{m-1} - 2^m f^*(x)$.

Proof: Lemma 1.12, Proposition 1.16 and Proposition 4.5. □

Before reading of Rothaus' work, the author was interested in the coding theory applications and so defined bentness using Proposition 1.10. He then used the function in Proposition 4.5 as a useful way to record the weights $|f + x^{\bar{\perp}}|$, and so came across the concept of duality by accident. This presentation involved a proof that f^* is bent using matrices and vectors indexed by the points of V — a matrix H with entries $h_{ij} = \omega^{i.j}$ to encode the RM_1 functions, and a vector indexed by the points of V to encode f. The fact that $f^{**} = f$ then followed from the fact that H is a symmetric Hadamard matrix, i.e. $HH^T = 2^n\mathbb{I}$.

We will see a further proof of this result in Chapter 9 (in the remarks following Theorem 9.9). However these proofs are all essentially the same as that of Proposition 1.5 — we must perform this counting somewhere.

Duality and equivalence

Proposition 4.7: *If $\alpha \in \mathrm{GL}(V)$ and $x \in V$ then*

(i) $\left(\overline{f}\right)^* = \overline{f^*}$.

(ii) $(f_x)^* = f^* + x^{\bar{\perp}}$.

(iii) $(f + x^{\bar{\perp}})^* = (f^*)_x$.

(iv) $(\alpha f)^* = \alpha^{-T} f^*$.

Proof:

(i) $\left(\overline{f}\right)^*(x) = 1 \iff |\overline{f} + x^{\bar{\perp}}| = |\overline{f + x^{\bar{\perp}}}|$ is high $\iff |f + x^{\bar{\perp}}|$ is low $\iff f^*(x) = 0$.

(ii) If $x.y = 0$ then

$$(f_x)^*(y) = 1 \iff |f_x + y^{\bar{\perp}}| = |f + (y^{\bar{\perp}})_x| = |f + y^{\bar{\perp}}| \text{ is high} \iff f^*(y) = 1$$

while if $x.y = 1$ then

$$\begin{aligned}(f_x)^*(y) = 1 &\iff |f_x + y^{\bar{\perp}}| = |f + (y^{\bar{\perp}})_x| = |f + y^{\perp}| \text{ is high} \\ &\iff |\overline{f + y^{\perp}}| = |f + y^{\bar{\perp}}| \text{ is low} \\ &\iff f^*(y) = 0\end{aligned}$$

so in either case $(f_x)^*(y) = f^*(y) + x.y = (f^* + x^{\bar{\perp}})(y)$.

(iii) Apply (ii) to f^* and take the dual of each side.

(iv)

$$\begin{aligned}(\alpha f)^*(x) = 1 &\iff |\alpha f + x^{\bar{\perp}}| \text{ is high} \\ &\iff |f + \alpha^{-1}(x^{\bar{\perp}})| = |f + (\alpha^T x)^{\bar{\perp}}| \text{ is high} \\ &\iff \alpha^T x \in f^* \iff x \in \alpha^{-T} f^* \iff \alpha^{-T} f^*(x) = 1.\end{aligned}$$

□

Corollary 4.8: $[\alpha, a, b^{(\sigma)}]f = g \iff [\alpha^{-T}, b, a^{(\sigma \,\hat{+}\, a.b)}]f^* = g^*$.

Proof: Using Proposition 4.7 we have

$$\begin{aligned}& g = [\alpha, a, b^{(\sigma)}]f = (\alpha f)_a + b^{(\sigma)} \\ \iff\ & g^* = \left((\alpha f)_a + b^{(\sigma)}\right)^* = \left(\alpha^{-T} f^* + a^{\bar{\perp}}\right)_b + 0^{(\sigma)} \\ & = (\alpha^{-T} f^*)_b + a^{(1 \,\hat{+}\, a.b)} + 0^{(\sigma)} = (\alpha^{-T} f^*)_b + a^{(\sigma \,\hat{+}\, a.b)} \\ & = [\alpha^{-T}, b, a^{(\sigma \,\hat{+}\, a.b)}]f^*.\end{aligned}$$

□

This suggests that we set $[\alpha, a, b^{(\sigma)}]^* \equiv [\alpha^{-T}, b, a^{(\sigma \hat{+} a.b)}]$, since then we have $\Big([\alpha, a, b^{(\sigma)}]f\Big)^* = [\alpha, a, b^{(\sigma)}]^* f^*$.

Thus we see that if f and g are equivalent then so are f^* and g^*. Hence if we can determine whether an equivalence class representative is equivalent to its dual, we know whether this is so for the whole class. Carrying this out for each of the class representatives with $n \leqslant 6$, we find that each of these class representatives is indeed equivalent to its dual, and hence for these n this is true of all bent functions. In the various classes we have representatives

$$
\begin{aligned}
&2.1: && f = x_1x_2\\
& && f^* = x_1x_2\\
&4.1: && f = x_1x_2 + x_3x_4\\
& && f^* = x_1x_2 + x_3x_4\\
&6.1: && f = x_1x_2 + x_3x_4 + x_5x_6\\
& && f^* = x_1x_2 + x_3x_4 + x_5x_6\\
&6.2: && f = x_1x_2x_3 + x_1x_4 + x_2x_5 + x_3x_6\\
& && f^* = x_4x_5x_6 + x_1x_4 + x_2x_5 + x_3x_6\\
&6.3: && f = x_1x_2x_5 + x_1x_3x_6 + x_1x_4 + x_2x_3 + x_2x_5 + x_5x_6\\
& && f^* = x_2x_4x_5 + x_3x_4x_6 + x_1x_4 + x_2x_3 + x_3x_6 + x_5x_6\\
&6.4: && f = x_1x_2x_3 + x_1x_5x_6 + x_2x_4x_6 + x_1x_3 + x_1x_4 + x_2x_3 + x_2x_4 + x_2x_5 + x_3x_6\\
& && f^* = x_1x_3x_5 + x_2x_3x_4 + x_4x_5x_6 + x_1x_4 + x_2x_4 + x_2x_5 + x_3x_6 + x_4x_6 + x_5x_6.
\end{aligned}
$$

Note, however, that because n is relatively small these functions are simpler than is generally the case. For example, in general a function and its dual need not have the same number of terms of each degree, as the functions above happen to do.

Also recall that it is possible for some of the functions in a class to be the same as their duals while others are not, as we saw above in the $n = 2$ case. Thus it is possible that other functions in classes 6.2, 6.3, 6.4 are the same as their duals, even though our chosen representatives are not. We will consider this question in much more detail in Chapter 14.

5. The four-function construction

Inducing lower-order bent functions

Consider a bent function f on $V = V(n, 2)$. If we pick a 2-dimensional subspace S of V then the cosets of S (its images under translation) partition V into sets of 4 points. If we write $V = R \oplus S$ then each coset is of the form $r + S$ for some $r \in R$, and we can use the points of R to index the cosets.

Now suppose that every coset contains either 1 or 3 points of (the support of) f. Then we can define a function g on R by

$$g(r) = \begin{cases} 1 & \text{if } r + S \text{ contains 3 points of } f \\ 0 & \text{if } r + S \text{ contains 1 point of } f. \end{cases}$$

Proposition 5.1: *With the above notation g is a bent function on R.*

Proof: We may assume that f is light, since if f induces g then $\overline{f}$ induces $\overline{g}$.

First,

$$3|g| + |\overline{g}| = |f| \implies 2|g| + 2^{n-2} = 2^{n-1} \pm 2^{m-1} \implies |g| = 2^{n-3} \pm 2^{m-2}$$

so g has the correct weight for a bent function.

Second, suppose we have some function $r^?$ in the 1st-order Reed-Muller code on R, where $r \neq 0$. Then the cosets of S in V corresponding to points in its support form a function in the 1st-order Reed-Muller code on V — in fact this function is just $r^?$, where r is now interpreted as a point of V.

Now let $\lambda = |g \cap r^?|$. Then λ of the cosets of S corresponding to points in $r^?$ contain 3 points of f, while the other $2^{n-3} - \lambda$ contain 1 point of f. But these points of f are exactly those in $f \cap r^?$. So, since f is bent, using Proposition 1.13 we have

$$\begin{aligned} 3\lambda + 2^{n-3} - \lambda &= 2^{n-2} \text{ or } 2^{n-2} - 2^{m-1} \\ \implies 2\lambda &= 2^{n-3} \text{ or } 2^{n-3} - 2^{m-1} \\ \implies \lambda &= 2^{n-4} \text{ or } 2^{n-4} - 2^{m-2}. \end{aligned}$$

Thus since $r^?$ was arbitrary we have shown that every $|g \cap r^?|$ has one of the two values in Proposition 1.13, so g is bent. □

For example, suppose we consider the bent function $f = x_1x_2 + x_1x_3 + x_3x_4$ on $V(4, 2)$. We can take $R = \langle e_1, e_2 \rangle$ and $S = \langle e_3, e_4 \rangle$. Then

$$\text{supp}(f) = \{11\cdot\cdot, 1\cdot1\cdot, 11\cdot1, \cdot\cdot11, \cdot111, 1111\}$$

and so

r	$r+S$	$\lvert f \cap (r+S)\rvert$	$g(r)$
$\cdot\cdot$	$\{\cdot\cdot\cdot\cdot, \cdot\cdot 1\cdot, \cdot\cdot\cdot 1, \cdot\cdot 11\}$	1	0
$1\cdot$	$\{1\cdot\cdot\cdot, 1\cdot 1\cdot, 1\cdot\cdot 1, 1\cdot 11\}$	1	0
$\cdot 1$	$\{\cdot 1\cdot\cdot, \cdot 11\cdot, \cdot 1\cdot 1, \cdot 111\}$	1	0
11	$\{11\cdot\cdot, 111\cdot, 11\cdot 1, 1111\}$	3	1

so that $g = x_1x_2$ is a bent function on R as expected.

At first sight this result is not very useful, since it appears to require a rather strict condition on f and then gives us a bent function of lower order than the one we started with. However, its value lies in the extent to which we are able to reverse the process.

When we pass from f to g we are throwing away information about the positioning of the 1 or 3 points of f within each coset of S. If we are to reverse this process, we must record the information we are throwing away. We do this by recording it in two functions p and q on R.

We fix a basis of S, and consider a coset $r + S$. Whether the coset contains 1 or 3 points of f there will be a point x which is different from the other 3 points — either x is in f and the other 3 are not or *vice versa*. The information we need to encode is the position of x within the coset, because this, together with $g(r)$ which tells us the number of points of f in the coset, allows us to reconstruct f.

The odd point out x in $r + S$ corresponds to a point $y = r + x$ in S. We define $p(r)$ to be the value of the first co-ordinate of y, and $q(r)$ to be the value of the second co-ordinate — these co-ordinates are taken with respect to our chosen basis of S.

If we repeat this process for all the cosets $r+S$ we will have defined two functions p and q on the whole of R, and they together with g allow us to reconstruct f. Going back to our previous example, and picking the obvious basis $\{e_3, e_4\}$ for S, we have

$$\mathrm{supp}(f) = \{11\cdot\cdot, 1\cdot 1\cdot, 11\cdot 1, \cdot\cdot 11, \cdot 111, 1111\}$$

and

r	$r+S$	Odd point x	y	$p(r)$	$q(r)$	$g(r)$
$\cdot\cdot$	$\{\cdot\cdot\cdot\cdot, \cdot\cdot 1\cdot, \cdot\cdot\cdot 1, \cdot\cdot 11\}$	$\cdot\cdot 11$	11	1	1	0
$1\cdot$	$\{1\cdot\cdot\cdot, 1\cdot 1\cdot, 1\cdot\cdot 1, 1\cdot 11\}$	$1\cdot 1\cdot$	$1\cdot$	1	0	0
$\cdot 1$	$\{\cdot 1\cdot\cdot, \cdot 11\cdot, \cdot 1\cdot 1, \cdot 111\}$	$\cdot 111$	11	1	1	0
11	$\{11\cdot\cdot, 111\cdot, 11\cdot 1, 1111\}$	$111\cdot$	$1\cdot$	1	0	1

So we encode f using the functions

$$g = x_1x_2, \qquad p = 1, \qquad q = x_1 + 1.$$

Conversely we can actually write down f in terms of g, p and q. Without loss of generality we can pick $S = \langle x_{n-1}, x_n \rangle$. Then we have the following:

Proposition 5.2: *With the above notation, and writing a point v of V as $r \mid s$ where $r \in R$, $s \in S$, we have*

$$f(r \mid s) = (x_{n-1} + p(r) + 1)(x_n + q(r) + 1) + g(r).$$

Proof: For cosets where $g(r) = 0$, a point is in f iff the value of its x_{n-1} co-ordinate agrees with $p(r)$, and similarly for x_n and $q(r)$. For cosets where $g(r) = 1$ we must take the complement of the coset, i.e. negate this condition. □

We can apply this to our example above. Recall that

$$g = x_1x_2, \qquad p = 1, \qquad q = x_1 + 1$$

so

$$\begin{aligned} f(r \mid s) &= (x_3 + 1 + 1)(x_4 + x_1 + 1 + 1) + x_1x_2 \\ &= x_3(x_4 + x_1) + x_1x_2 = x_1x_2 + x_1x_3 + x_3x_4 \end{aligned}$$

as expected.

The important thing about p and q is that they give us three more bent functions on R:

Proposition 5.3: *With the above notation $g + p$, $g + q$ and $g + p + q$ are bent functions on R.*

Proof: p (respectively q, $p + q$) indicates the position of the odd point out with respect to the vector e_{n-1} (e_n, $e_{n-1} + e_n$) in S. Since we could have chosen any of these 3 vectors as the first of our basis vectors it is enough to prove the result for $h = g + p$, say.

As before we may assume that f is light, since complementing f does not alter the position of the odd point out in each coset, so leaves p and q unchanged.

Pick some function $r^{\bar{\perp}}$ in the 1st-order Reed-Muller code on R. Consider a coset $t + S$. This contains 4 points all of which are in the same one of $r^{\bar{\perp}}$ or $r^{\perp}$, but 2 of which are in each of ${e_{n-1}}^{\bar{\perp}}$ and ${e_{n-1}}^{\perp}$. Thus 2 of its points are in $(r + e_{n-1})^{\bar{\perp}}$ and the other 2 are not.

Each of these 2 pairs of points in $t+S$ may contain some points of f. We will calculate the *difference* between the number of points of f in the 2 pairs, i.e. we will calculate

$$|f \cap (t+S) \cap (r+e_{n-1})^{\bar{\perp}}| - |f \cap (t+S) \cap (r+e_{n-1})^{\perp}|. \qquad (*)$$

There are eight cases, according to the values of $g(r)$ and $p(r)$ and whether $t+S \subseteq r^{\bar{\perp}}$ or not. For example, if $g(t)=p(t)=0$ and $t+S \subseteq r^{\bar{\perp}}$ then (using g) $t+S$ contains 1 point of f and (using p) this point is not in $e_{n-1}{}^{\bar{\perp}}$, so it *is* in $(r+e_{n-1})^{\bar{\perp}}$. Thus in $(*)$ we have 1 point in the first set and 0 in the second so the required difference is $1-0=1$.

Carrying out this analysis for all eight cases we obtain

$t+S \subseteq r^{\bar{\perp}}$	$g(t)$	$p(t)$	$h(t)$	# points in first set	# points in second set	Difference
Yes	0	0	0	1	0	$+1$
Yes	0	1	1	0	1	-1
Yes	1	0	1	1	2	-1
Yes	1	1	0	2	1	$+1$
No	0	0	0	0	1	-1
No	0	1	1	1	0	$+1$
No	1	0	1	2	1	$+1$
No	1	1	0	1	2	-1

If we consider 1 to correspond to truth and 0 to falsehood, then we see that the difference is -1 precisely when $h(t)$ agrees with $t+S \subseteq r^{\bar{\perp}}$. In other words the -1s indicate points in the intersection of (the support of) h and $r^{\bar{\perp}}$ and in the intersection of their complements $\overline{h}$ and $r^{\perp}$. So the number of -1s is

$$|h \cap r^{\bar{\perp}}| + |\overline{h} \cap r^{\perp}| = |h \cap r^{\bar{\perp}}| + (|r^{\perp}| - |h \cap r^{\perp}|) = |h \cap r^{\bar{\perp}}| + 2^{n-3} - |h \cap r^{\perp}|.$$

Similarly the $+1$s indicate points in $h \cap r^{\perp}$ and $\overline{h} \cap r^{\bar{\perp}}$, so the number of $+1$s is

$$|h \cap r^{\perp}| + 2^{n-3} - |h \cap r^{\bar{\perp}}|.$$

So if we add the differences over all cosets $t+S$ we will obtain

$$-(|h \cap r^{\bar{\perp}}| + 2^{n-3} - |h \cap r^{\perp}|) + (|h \cap r^{\perp}| + 2^{n-3} - |h \cap r^{\bar{\perp}}|) = 2(|h \cap r^{\perp}| - |h \cap r^{\bar{\perp}}|).$$

On the other hand, in adding the differences we will, by $(*)$, obtain $|f \cap (r+e_{n-1})^{\bar{\perp}}| - |f \cap (r+e_{n-1})^{\perp}|$. Now, since f is bent and $(r+e_{n-1})^{\bar{\perp}} \in \mathrm{RM}_1$, by Corollary 4.6(iv) this must be $\pm 2^{m-1}$. So

$$2(|h \cap r^{\perp}| - |h \cap r^{\bar{\perp}}|) = \pm 2^{m-1} \implies |h \cap r^{\perp}| - |h \cap r^{\bar{\perp}}| = \pm 2^{m-2}.$$

But $r^{\bar{\perp}}$ was arbitrary, so by Proposition 1.16 we have checked that h is bent. $\square$

The four-function construction

Our proof that g was bent used a quarter of our conditions on f — those involving functions of the form $r^{\bar{\perp}}$ with $r \in R$. Our proof for $g+p$, $g+q$ and $g+p+q$ involved the other three-quarters — functions of the form $(r+e_{n-1})^{\bar{\perp}}$, $(r+e_n)^{\bar{\perp}}$ and $(r+e_{n-1}+e_n)^{\bar{\perp}}$. Thus we might expect that if we know that these four functions *are* bent, this is enough to ensure that f is also. This is indeed the case:

Theorem 5.4: *With the above notation, if g, p and q are such that g, $g+p$, $g+q$ and $g+p+q$ are bent functions then the (unique) function f specified by them is also bent.*

Proof: Any function in RM_1 can be written as $(r+s)^{\bar{\perp}}$ for some $r \in R$, $s \in S$. We set $h = g$, $g+p$, $g+q$, $g+p+q$ according as $s = 0$, x_{n-1}, x_n, $x_{n-1}+x_n$, and count the difference

$$|f \cap (t+S) \cap (r+s)^{\bar{\perp}}| - |f \cap (t+S) \cap (r+s)^{\perp}|$$

for all cosets $t+S$. As before, we obtain

$$2(|h \cap r^{\perp}| - |h \cap r^{\bar{\perp}}|) = |f \cap (r+s)^{\bar{\perp}}| - |f \cap (r+s)^{\perp}|.$$

Now since h is bent the LHS of this equation is $\pm 2.2^{m-2} = \pm 2^{m-1}$, hence so is the RHS. Thus as before since $r+s$ was arbitrary we have checked that f is bent. □

This construction is Rothaus' construction II [35].

Noting that we can write $g+p+q = g+(g+p)+(g+q)$, we can rewrite this result using Proposition 5.2, our explicit expression for f:

Corollary 5.5 — The four-function construction: *Suppose that g, h, k are bent functions on a vector space $R = \langle e_1, \ldots, e_{n-2} \rangle$ such that $g+h+k$ is bent. Then*

$$f = (x_{n-1}+g+h+1)(x_n+g+k+1)+g$$

is a bent function on $R \oplus \langle e_{n-1}, e_n \rangle$.

Proof: From Proposition 5.2 and Theorem 5.4, setting $h = g+p$, $k = g+q$. □

We call this the *four-function construction* since it involves finding four bent functions on R which sum to zero — these are the functions we called g, h, k and $g+h+k$ above. It is mentioned by Rothaus [35], and also in [30] and [40], although no derivation such as the above is given.

However, note that (so far) the four-function construction is not explicit, since it requires us to search for an appropriate set of four functions and we have not yet shown that any such set exists for all even n, let alone how to find one. Also, we shall see in Chapter 9 that the proportion of bent functions which can be produced even by this new construction seems rather small.

Another way of representing the four-function construction

In order to view the four-function construction more graphically, we can represent the points of the vector space $V = R \oplus S$ in a $2^{n-2} \times 4$ matrix, indexing the rows and columns by R and S so that the (r, s)th entry is $f(r \mid s)$.

For example if we take $R = \langle e_1, e_2 \rangle$, $S = \langle e_3, e_4 \rangle$ and

$$f = x_1x_2 + x_1x_3 + x_3x_4 \quad \Longrightarrow \quad \mathrm{supp}(f) = \{11\cdot\cdot, 1\cdot1\cdot, 11\cdot1, \cdot\cdot11, \cdot111, 1111\}$$

as before then we get the matrix

		s →			
		$\cdot\cdot$	$1\cdot$	$\cdot1$	11
	$\cdot\cdot$	$\cdot$	$\cdot$	$\cdot$	1
r	$1\cdot$	$\cdot$	1	$\cdot$	$\cdot$
↓	$\cdot1$	$\cdot$	$\cdot$	$\cdot$	1
	11	1	$\cdot$	1	1

Now we can plot the four functions involved in the construction on the columns of a similar matrix, the *four-function matrix*, as follows — recall that

$$g = x_1x_2, \qquad g+p = x_1x_2+1, \qquad g+q = x_1x_2+x_1+1, \qquad g+p+q = x_1x_2+x_1$$

so we have

		g	$g+p$	$g+q$	$g+p+q$
	$\cdot\cdot$	$\cdot$	1	1	$\cdot$
r	$1\cdot$	$\cdot$	1	$\cdot$	1
↓	$\cdot1$	$\cdot$	1	1	$\cdot$
	11	1	$\cdot$	1	$\cdot$

Now if we write f in terms of g, p and q using Proposition 5.2, i.e.

$$f(r \mid s) = (x_{n-1} + p(r) + 1)(x_n + q(r) + 1) + g(r),$$

we see that the values $f(r \mid \cdot)$ of f for some fixed r depend only on the values of g, p and q at that same value of r. In other words, the rth row of the f matrix depends only on the rth row of the four-function matrix.

So we can write down a conversion table relating rows of the two matrices. Note that in this table the rows do not correspond to points in R — instead we consider all possible rows. In fact since the four functions $g, g+p, g+q, g+p+q$ sum to $\mathbb{O}$ we need consider only rows with even weight in the four-function matrix:

g	$g+p$	$g+q$	$g+p+q$		p	q		s ↓ $\cdot\cdot$	$1\cdot$	$\cdot 1$	11
·	·	·	·		·	·		1	·	·	·
1	1	·	·		·	1		1	1	·	1
1	·	1	·		1	·		1	·	1	1
·	1	1	·	$\Longrightarrow$	1	1	$\Longrightarrow$	·	·	·	1
1	·	·	1		1	1		1	1	1	·
·	1	·	1		1	·		·	1	·	·
·	·	1	1		·	1		·	·	1	·
1	1	1	1		·	·		·	1	1	1

At first sight there does not seem to be a simple relationship between the rows. However, we observe that every row of the f matrix (on the right above) contains oddly many 1s, so represents the support of a bent function on S — recall that the bent functions in the case $n = 2$ are precisely those of odd weight.

We can therefore calculate the dual of each row of the f matrix as a function on S. We find that in each row the dual is closely related to the corresponding row of the four-function matrix — they always agree in the first column and disagree in the others. In other words if we consider the complements of all but the first of the four functions, and label the columns of the four-function matrix with points of S, we obtain two matrices with the property that corresponding rows are duals of each other:

Points of $S \to$	g $\cdot\cdot$	$\overline{g+p}$ $1\cdot$	$\overline{g+q}$ $\cdot 1$	$\overline{g+p+q}$ 11		$\cdot\cdot$	$1\cdot$	$\cdot 1$	11
	·	1	1	1		1	·	·	·
	1	·	1	1		1	1	·	1
	1	1	·	1		1	·	1	1
	·	·	·	1	dualise rows ⟶	·	·	·	1
	1	1	1	·		1	1	1	·
	·	·	1	·		·	1	·	·
	·	1	·	·		·	·	1	·
	1	·	·	·		·	1	1	1

We can now express our construction of f from g, p and q in another form:

Theorem 5.6 — The four-function construction: *Suppose that g, h, k are bent functions on a vector space $R = \langle e_1, \ldots, e_{n-2} \rangle$ such that $g + h + k$ is bent. Write*

down the supports of g, h, k and $\overline{g+j+k}$ as columns of a $2^{n-2} \times 4$ matrix and identify these columns with the points of $S = \langle e_{n-1}, e_n \rangle$. Each row of this matrix will have odd weight, so consider the rows as supports of bent functions on S and replace each row by its dual. Let f be the function represented by the new matrix. Then f is a bent function on $R \oplus S$.

Proof: From the above discussion, but note that this time h corresponds to $\overline{g+p}$ (rather than $g+p$ as in Corollary 5.5), and similarly for k. □

So consider our example. We take

$$g = x_1x_2, \qquad h = \overline{g+p} = x_1x_2, \qquad k = \overline{g+q} = x_1x_2 + x_1$$

and so obtain the matrices

		g	h	k	$\overline{g+h+k}$					
		$\cdot\cdot$	$1\cdot$	$\cdot 1$	11			$\cdot\cdot$	$1\cdot$	$\cdot 1$	11
	$\cdot\cdot$	$\cdot$	$\cdot$	$\cdot$	1			$\cdot$	$\cdot$	$\cdot$	1
r	$1\cdot$	$\cdot$	$\cdot$	1	$\cdot$		dualise rows $\longrightarrow$	$\cdot$	1	$\cdot$	$\cdot$
$\downarrow$	$\cdot 1$	$\cdot$	$\cdot$	$\cdot$	1			$\cdot$	$\cdot$	$\cdot$	1
	11	1	1	$\cdot$	1			1	$\cdot$	1	1

This second matrix represents $f = x_1x_2 + x_1x_3 + x_3x_4$ as expected.

For a more complicated example, we use this construction to obtain a function of class 6.3. The function we want to construct is $f = x_1x_2x_5 + x_1x_3x_6 + x_1x_4 + x_2x_3 + x_2x_5 + x_5x_6$ (see Chapter 3). We take

$$g = x_1x_4 + x_2x_3, \qquad h = x_1x_3 + x_1x_4 + x_2x_3, \qquad k = x_1x_2 + x_1x_4 + x_2x_3 + x_2.$$

Then we construct the matrices as follows:

		g	h	k	$\overline{g+h+k}$					
		$\cdot\cdot$	$1\cdot$	$\cdot 1$	11			$\cdot\cdot$	$1\cdot$	$\cdot 1$	11
	$\cdot\cdot\cdot\cdot$	$\cdot$	$\cdot$	$\cdot$	1			$\cdot$	$\cdot$	$\cdot$	1
	$1\cdot\cdot\cdot$	$\cdot$	$\cdot$	$\cdot$	1			$\cdot$	$\cdot$	$\cdot$	1
	$\cdot 1\cdot\cdot$	$\cdot$	$\cdot$	1	$\cdot$			$\cdot$	1	$\cdot$	$\cdot$
	$11\cdot\cdot$	$\cdot$	$\cdot$	$\cdot$	1			$\cdot$	$\cdot$	$\cdot$	1
	$\cdot\cdot 1\cdot$	$\cdot$	$\cdot$	$\cdot$	1			$\cdot$	$\cdot$	$\cdot$	1
	$1\cdot 1\cdot$	$\cdot$	1	$\cdot$	$\cdot$			$\cdot$	$\cdot$	1	$\cdot$
	$\cdot 11\cdot$	1	1	$\cdot$	1			1	$\cdot$	1	1
r	$111\cdot$	1	$\cdot$	1	1			1	1	$\cdot$	1
$\downarrow$	$\cdot\cdot\cdot 1$	$\cdot$	$\cdot$	$\cdot$	1	$\xrightarrow{\text{dualise rows}}$		$\cdot$	$\cdot$	$\cdot$	1
	$1\cdot\cdot 1$	1	1	1	$\cdot$			1	1	1	$\cdot$
	$\cdot 1\cdot 1$	$\cdot$	$\cdot$	1	$\cdot$			$\cdot$	1	$\cdot$	$\cdot$
	$11\cdot 1$	1	1	1	$\cdot$			1	1	1	$\cdot$
	$\cdot\cdot 11$	$\cdot$	$\cdot$	$\cdot$	1			$\cdot$	$\cdot$	$\cdot$	1
	$1\cdot 11$	1	$\cdot$	1	1			1	1	$\cdot$	1
	$\cdot 111$	1	1	$\cdot$	1			1	$\cdot$	1	1
	1111	$\cdot$	1	$\cdot$	$\cdot$			$\cdot$	$\cdot$	1	$\cdot$

and it is straightforward to check that this second matrix represents f as expected. The construction can also be checked using Proposition 5.2 — we calculate

$$p = \overline{g+h} = (x_1x_4 + x_2x_3) + (x_1x_3 + x_1x_4 + x_2x_3) + 1 = x_1x_3 + 1$$
$$q = \overline{g+k} = (x_1x_4 + x_2x_3) + (x_1x_2 + x_1x_4 + x_2x_3 + x_2) + 1 = x_1x_2 + x_2 + 1$$

and hence

$$\begin{aligned} f &= (x_5 + x_1x_3 + 1 + 1)(x_6 + x_1x_2 + x_2 + 1 + 1) + x_1x_4 + x_2x_3 \\ &= x_5x_6 + x_1x_2x_5 + x_2x_5 + x_1x_3x_6 + x_1x_2x_3 + x_1x_2x_3 + x_1x_4 + x_2x_3 \\ &= x_1x_2x_5 + x_1x_3x_6 + x_1x_4 + x_2x_3 + x_2x_5 + x_5x_6 \end{aligned}$$

as required.

6. Generalisation of the four-function construction

Initially in Chapter 5 the point about our four bent functions g, h, k and $\overline{g+h+k}$ was that they summed to $\mathbb{1}$ so that all the rows of the four-function matrix had odd weight. However, in carrying out the construction in Theorem 5.6 the only fact about the rows which we actually used was that each represented a bent function on S.

This suggests that we might be able to extend the construction to cases other than that of $\dim S = 2$. We start by generalising the matrices used at the end of Chapter 5:

Definition: If R, S are $\mathbb{F}_2$-vector spaces a *matrix indexed by* $R \oplus S$ is a matrix of 0s and 1s with its rows indexed by the points of R and columns indexed by the points of S.

The matrix $A = (a_{rs})$ *represents* a function f on $R \oplus S$ if $f(r \mid s) = a_{rs}$. □

We find that the four-function construction (Theorem 5.6) does indeed generalise in a very straightforward way:

Theorem 6.1 — The direct-summand construction: *Let A be a matrix indexed by $R \oplus S$ with $\dim R = n_r = 2m_r$, $\dim S = n_s = 2n_s$ both even. Suppose A has the property that each row $a_{r\bullet}$ represents a bent function of S and each column $a_{\bullet s}$ represents a bent function of R. Let B be the matrix whose rth row represents the dual of the rth row of A. Then B represents a bent function f on $R \oplus S$.*

Proof: Let $V = R \oplus S$ and let $(\rho+\sigma)^{\bar{\perp}}$ be a function on V in RM_1. We need to check $|f + (\rho+\sigma)^{\bar{\perp}}|$ for each $\rho \in R$, $\sigma \in S$.

Summing over the rows of B we have

$$|f + (\rho+\sigma)^{\bar{\perp}}| = \sum_{x \in R} |b_{x\bullet} + (\rho+\sigma)^{\bar{\perp}}|.$$

Now considering functions on S

$$|b_{x\bullet} + (\rho+\sigma)^{\bar{\perp}}| = \begin{cases} |b_{x\bullet} + \sigma^{\bar{\perp}}| & \text{if } x.\rho = 0 \\ 2^{n_s} - |b_{x\bullet} + \sigma^{\bar{\perp}}| & \text{if } x.\rho = 1 \end{cases}$$

and by Corollary 4.6(ii) and the definition of B

$$\begin{aligned} |b_{x\bullet} + \sigma^{\bar{\perp}}| &= 2^{n_s-1} - 2^{m_s-1} + 2^{m_s} {b_{x\bullet}}^*(\sigma) \\ &= 2^{n_s-1} - 2^{m_s-1} + 2^{m_s} a_{x\sigma}. \end{aligned}$$

So

$$\begin{aligned}|f+\sigma^{\bar\perp}| &= \sum_{x\in R}|b_{x\bullet}+\sigma^{\bar\perp}|\\ &= \sum_{x\in R}(2^{n_s-1}-2^{m_s-1}+2^{m_s}a_{x\sigma})\\ &= 2^{n_r}2^{n_s-1}-2^{n_r}2^{m_s-1}+2^{m_s}\sum_{x\in R}a_{x\sigma}\\ &= 2^{n-1}-2^{n_r+m_s-1}+2^{m_s}|a_{\bullet\sigma}|\\ &= 2^{n-1}-2^{n_r+m_s-1}+2^{m_s}(2^{n_r-1}\pm 2^{m_r-1})\\ &= 2^{n-1}\pm 2^{m-1}.\end{aligned}$$

Similarly if $\rho\neq 0$ then using Lemma 1.15(i) we have

$$\begin{aligned}|f+(\rho+\sigma)^{\bar\perp}| &= \sum_{x\in R}|b_{x\bullet}+(\rho+\sigma)^{\bar\perp}|\\ &= \sum_{x\in\rho^\perp}(2^{n_s-1}-2^{m_s-1}+2^{m_s}a_{x\sigma})\\ &\qquad +\sum_{x\in\rho^{\bar\perp}}(2^{n_s}-2^{n_s-1}+2^{m_s-1}-2^{m_s}a_{x\sigma})\\ &= 2^{n_r-1}(2^{n_s-1}-2^{m_s-1})+2^{m_s}\sum_{x\in\rho^\perp}a_{x\sigma}\\ &\qquad +2^{n_r-1}(2^{n_s-1}+2^{m_s-1})-2^{m_s}\sum_{x\in\rho^{\bar\perp}}a_{x\sigma}\\ &= 2.2^{n_r-1}2^{n_s-1}+2^{m_s}(|a_{\bullet\sigma}\cap\rho^\perp|-|a_{\bullet\sigma}\cap\rho^{\bar\perp}|)\\ &= 2^{n_r+n_s-1}+2^{m_s}(|a_{\bullet\sigma}+\rho^{\bar\perp}|-2^{n_r-1})\\ &= 2^{n-1}+2^{m_s}(2^{n_r-1}\pm 2^{m_r-1}-2^{n_r-1})\\ &= 2^{n-1}\pm 2^{m-1}\end{aligned}$$

so, since ρ and σ were arbitrary, f is bent by Proposition 1.10. □

A converse to this result holds as well:

Theorem 6.2: *Let f be a bent function on $V=R\oplus S$ and let A be the matrix representing f on $R\oplus S$, with R and S of even dimension. If the rows of A represent bent functions of S then let B be the matrix whose rows represent the duals of these rows. Then the columns of B represent bent functions on R.*

Proof: Consider $b_{\bullet s}$, the sth column of B. Let $\rho^{\bar\perp}$ be a function in RM_1 on R. Let n_r, etc. be as in Theorem 6.1. We need to check $|b_{\bullet s}+\rho^{\bar\perp}|$ for each $\rho\in R$.

By the definition of B and Corollary 4.6

$$\begin{aligned}b_{xs} &= a_{x\bullet}{}^*(s) = 2^{-m_s}(|a_{x\bullet}+s^{\bar\perp}|-2^{n_s-1}+2^{m_s-1})\\ &= 2^{-m_s}|a_{x\bullet}+s^{\bar\perp}|-2^{m_s-1}+2^{-1}\end{aligned}$$

and so

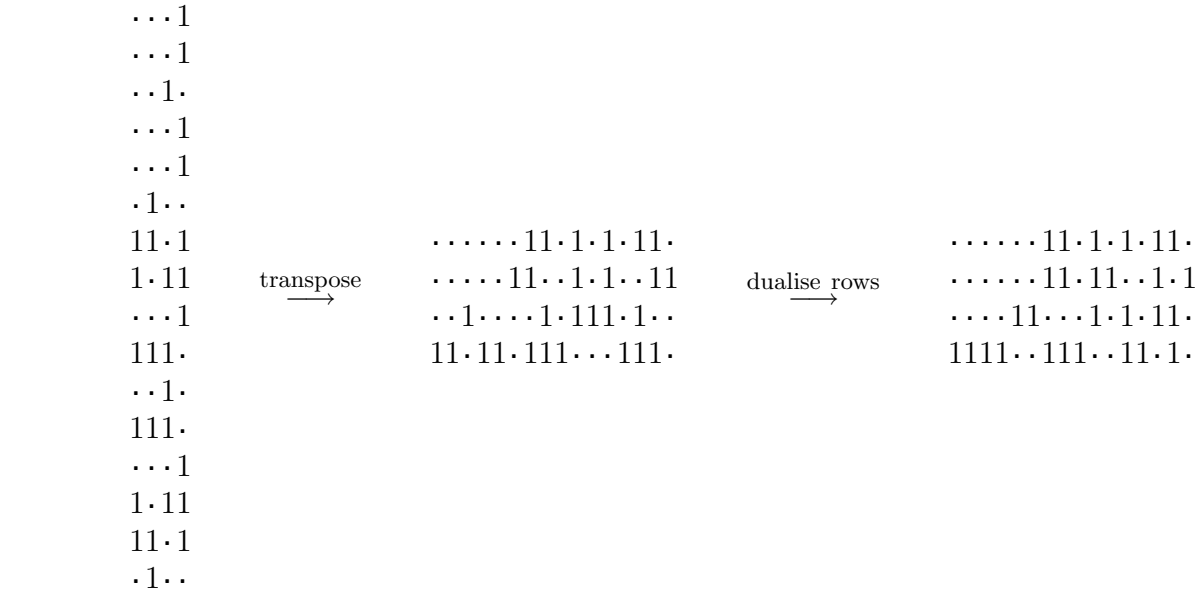
$$\begin{aligned}
|b_{\bullet s}| &= \sum_{x\in R} b_{xs} \\
&= \sum_{x\in R} (2^{-m_s}|a_{x\bullet} + s^{\bar{\perp}}| - 2^{m_s-1} + 2^{-1}) \\
&= 2^{-m_s} \sum_{x\in R} |a_{x\bullet} + s^{\bar{\perp}}| - 2^{n_r+m_s-1} + 2^{n_r-1} \\
&= 2^{-m_s}|f + s^{\bar{\perp}}| - 2^{n_r+m_s-1} + 2^{n_r-1} \\
&= 2^{-m_s}(2^{n-1} \pm 2^{m-1}) - 2^{n_r+m_s-1} + 2^{n_r-1} \\
&= 2^{n_r-1} \pm 2^{m_r-1}.
\end{aligned}$$

Similarly if $\rho \neq 0$ then $|b_{\bullet s} \cap \rho^{(\sigma)}| = 2^{-m_s}|(f + s^{\bar{\perp}}) \cap \rho^{(\sigma)}| - 2^{n_r+m_s-1} + 2^{n_r-1}$ and so by Lemma 1.15(i) (twice)

$$\begin{aligned}
|b_{\bullet s} + \rho^{\bar{\perp}}| &= |b_{\bullet s} \cap \rho^{\bar{\perp}}| - |b_{\bullet s} \cap \rho^{\perp}| + 2^{n_r-1} \\
&= 2^{-m_s}|(f + s^{\bar{\perp}}) \cap \rho^{\bar{\perp}}| - 2^{n_r+m_s-1} + 2^{n_r-1} \\
&\qquad - 2^{-m_s}|(f + s^{\bar{\perp}}) \cap \rho^{\perp}| + 2^{n_r+m_s-1} - 2^{n_r-1} + 2^{n_r-1} \\
&= 2^{-m_s}(|(f + s^{\bar{\perp}}) + \rho^{\bar{\perp}}| - 2^{n-1}) + 2^{n_r-1} \\
&= 2^{-m_s}(2^{n-1} \pm 2^{m-1} - 2^{n-1}) + 2^{n_r-1} \\
&= 2^{n_r-1} \pm 2^{m_r-1}.
\end{aligned}$$

so, since ρ was arbitrary, $b_{\bullet s}$ is bent by Proposition 1.10. □

As an example of the direct-summand construction, consider the matrix indexed by $R \oplus S$ used in the $f = x_1x_2x_5 + x_1x_3x_6 + x_1x_4 + x_2x_3 + x_2x_5 + x_5x_6$ example in Chapter 5 — call this matrix M, say. By construction the columns of M are bent functions, so the rows of M^T are bent functions, so we can apply Theorem 6.1 to M^T as follows:



By adding polynomials with singleton support we find that this last matrix represents the function

$$g = x_1x_4x_6 + x_2x_5x_6 + x_1x_2 + x_2x_5 + x_3x_6 + x_4x_5$$

and it is a straightforward (if lengthy) check that this function is bent.

Thus although Theorems 6.1 and 6.2 involve dualising the rows of a matrix we could also dualise its columns. So from a matrix indexed by $R \oplus S$ whose rows and columns all represent bent functions we obtain two bent functions on $V = R \oplus S$. However in fact these functions are actually duals as bent functions on V:

Theorem 6.3: *Let A be a matrix indexed by $R \oplus S$ whose rows and columns are bent functions. Let B (respectively C) be the matrix obtained by dualising the rows (columns) of A and let b (c) be the function on V represented by B (C). Then $b^* = c$.*

Proof: If $r = 0$ we have

$$\begin{aligned}
|b + s^{\bar{\perp}}| &= \sum_{x \in R} |a_{x\bullet}{}^* + s^{\bar{\perp}}| \\
&= \sum_{x \in R} (2^{n_s-1} - 2^{m_s-1} + 2^{m_s} a_{xs}) \\
&= 2^{n_r} 2^{n_s-1} - 2^{n_r} 2^{m_s-1} + 2^{m_s} |a_{\bullet s}| \\
&= 2^{n-1} - 2^{n_r+m_s-1} + 2^{m_s}(2^{n_r-1} - 2^{m_r-1} + 2^{m_r} a_{\bullet s}{}^*(0)) \\
&= 2^{n-1} - 2^{m-1} + 2^m a_{\bullet s}{}^*(0)
\end{aligned}$$

while if $r \neq 0$ we have

$$\begin{aligned}
|b + (r \mid s)^{\bar{\perp}}| &= \sum_{x \in R} |a_{x\bullet}{}^* + r^{\bar{\perp}} + s^{\bar{\perp}}| \\
&= \sum_{x \in r^{\bar{\perp}}} \left(|S| - |a_{x\bullet}{}^* + s^{\bar{\perp}}| \right) + \sum_{x \in r^{\perp}} |a_{x\bullet}{}^* + s^{\bar{\perp}}| \\
&= 2^{n_r-1} 2^{n_s} - \sum_{x \in r^{\bar{\perp}}} |a_{x\bullet}{}^* + s^{\bar{\perp}}| + \sum_{x \in r^{\perp}} |a_{x\bullet}{}^* + s^{\bar{\perp}}| \\
&= 2^{n-1} - \sum_{x \in r^{\bar{\perp}}} \left(2^{n_s-1} - 2^{m_s-1} + 2^{m_s} a_{xs} \right) \\
&\qquad + \sum_{x \in r^{\perp}} \left(2^{n_s-1} - 2^{m_s-1} + 2^{m_s} a_{xs} \right) \\
&= 2^{n-1} - 2^{m_s} \sum_{x \in r^{\bar{\perp}}} a_{xs} + 2^{m_s} \sum_{x \in r^{\perp}} a_{xs} \\
&= 2^{n-1} - 2^{m_s} \left(|a_{\bullet s} \cap r^{\bar{\perp}}| - |a_{\bullet s} \cap r^{\perp}| \right) \\
&= 2^{n-1} - 2^{m_s} (2^{m_r-1} - 2^{m_r} a_{\bullet s}{}^*(r)) \\
&= 2^{n-1} - 2^{m-1} + 2^m a_{\bullet s}{}^*(r).
\end{aligned}$$

Thus in either case, using Corollary 4.6(ii), $b^*(r \mid s) = a_{\bullet s}{}^*(r) = c(r \mid s)$. □

Returning to our example, recall that we obtained the function

$$g = x_1x_4x_6 + x_2x_5x_6 + x_1x_2 + x_2x_5 + x_3x_6 + x_4x_5$$

by dualising the rows of M^T. The function obtained by dualising the columns of M can be found from g by relabelling the variables to undo the effect of the transposition. After relabelling (and sorting the terms) we get the function

$$h = x_2x_4x_5 + x_3x_4x_6 + x_1x_4 + x_2x_3 + x_3x_6 + x_5x_6$$

and this is indeed the dual of

$$f = x_1x_2x_5 + x_1x_3x_6 + x_1x_4 + x_2x_3 + x_2x_5 + x_5x_6$$

as expected.

After we dualise the rows of our matrix indexed by $R \oplus S$ the columns may no longer represent bent functions (although clearly the rows will still do so). Equivalently, the function represented by the original matrix indexed by $R \oplus S$ need not be bent. For example if $n = 4$ and the matrix is $\mathbb{I}_4$ then all its rows and columns are bent, but the function it represents has weight 4 so cannot be bent.

On the other hand it is *possible* for the function represented by the matrix indexed by $R \oplus S$ to be bent. For example if we use the matrix

$$\begin{matrix} \cdots 1 \\ \cdots 1 \\ \cdots 1 \\ 111\cdot \end{matrix}$$

then every row is self-dual so this *is* the f matrix, and represents the bent function $f = x_1x_2 + x_3x_4$.

Ways of forming suitable matrices

So far we have seen only how to get from a matrix with all its rows and columns representing bent functions to another bent function — we have not seen how to find such matrices in the first place. In fact this simultaneous condition on the rows and columns is quite hard to satisfy. However, there are various special cases which are easier to work with, several of which have been studied as constructions in their own right — see [30] or [40] (Methods 1 and 2) and [23] or [35] (Method 4), for example.

Method 1 For $n_s = 2$ we merely have to find four bent functions g_1, g_2, g_3, g_4 summing to $\mathbb{O}$ so that we can use the construction of Corollary 5.5. The simplest way to do this is to take $g_1 = g_2 = g_3 = g_4 = g$, so that every row of our matrix for f is either $\cdots 1$ or $111\cdot$. By Corollary 5.5 we have

$$f = (x_{n-1} + 1)(x_n + 1) + g.$$

Method 2 More generally, we can take $g_1 = g_2$ and $g_3 = g_4$. By Corollary 5.5 we have

$$f = (x_{n-1} + 1)(x_n + g_1 + g_3 + 1) + g_1.$$

Method 3 Another method of finding suitable g_is involves translates of sets in V, and we discuss these in some detail. Consider translating a function f on V through a vector v. Given f as a polynomial in co-ordinates $\{x_1, \ldots, x_n\}$ with respect to a basis $\{e_1, \ldots, e_n\}$ of V we can calculate the polynomial f_v quite easily. As described earlier we replace x_i by $(x_i + 1)$ throughout f iff $v.e_i = 1$. Since translation through v complements the value of each such x_i the value of the resultant polynomial at a point $x + v$ is clearly equal to that of the original polynomial at x.

For example if $n = 2$ and f is the set $\{\cdot\cdot, 11\}$, i.e. the polynomial $x_1 + x_2 + 1$, then the translate of f by $v = 1\cdot$ has support $\{1\cdot, \cdot 1\}$ which is the support of $(x_1 + 1) + x_2 + 1 = x_1 + x_2$.

Proposition 6.4: *$f_{\langle v \rangle}$ has lower degree than f.*

Proof: f_v consists of f together with the extra terms caused by multiplying out the various $(x_i + 1)$s introduced into f by the translation. Since each of these extra terms includes at least one 1 instead of the x_i in the corresponding original term, its degree must be less than that of the original term, hence less than that of f.

Thus the highest degree terms of f and f_v are identical, so when we add them to produce $f_{\langle v \rangle}$ these highest degree terms all cancel out, so producing a function of lower degree than f as claimed. □

Corollary 6.5: *A non-singular quadratic function f is bent.*

Proof: By adding a suitable RM_1 function we may assume that f is homogeneous. Now pick $0 \neq v \in V$ and consider $g = f_{\langle v \rangle}$. By Proposition 6.4 its degree is lower than $\deg f = 2$. On the other hand if its degree is 0 then g is $\emptyset$ or V so we must have $|f| = 2^{n-1}$, whereas a non-singular homogeneous quadratic has weight $2^{n-1} - 2^{m-1}$ by Corollary 3.11. Thus we must have $\deg g = 1$ and hence $|g| = 2^{n-1}$. But now since v was arbitrary we have checked that $|f_{\langle v \rangle}| = 2^{n-1}$ for all non-zero v, so f is bent by Theorem 1.19. □

Corollary 6.6: *A quadratic function f of rank $2r$ on $V(n,2)$ has weight 2^{n-1} or $2^{n-1} \pm 2^{n-r-1}$.*

Proof: Let q denote the homogeneous quadratic portion of f, i.e. the degree 2 terms, and write $f = q + l^?$ where $l^? \in \mathrm{RM}_1$. Without loss of generality $q = x_1x_2 + \ldots + x_{2r-1}x_{2r}$ where $2r$ is the rank of q.

So let $W = \langle x_1, \ldots, x_{2r} \rangle$ so that q is a bent function on W. Now either $l^?$ is the same function $m^?$ on each coset of W, in which case $|f|$ is just $2^{n-2r}|q + l^?| = 2^{n-1} \pm 2^{n-r-1}$, or $l^?$ is $m^?$ on half the cosets of W and $\overline{m^?}$ on the other half, in which case $|f|$ is 2^{n-1}. □

Note that any function g equivalent to a non-singular quadratic f is also a non-singular quadratic, for we shall see in Chapter 8 that g must have the same degree as f, and if g is singular we can add an RM_1 function to it to produce a singular homogeneous quadratic which would then have the wrong weight to be bent, by Corollary 3.11.

We define a special term for the extreme case of Proposition 6.4's degree-lowering effect:

Definition: A vector v is a *stabiliser* of a set $f \subseteq V$ if $f + f_v = \emptyset$. □

This is equivalent to saying that $x \in f \iff x + v \in f$ for all $x \in V$. Note that f and $\overline{f}$ have the same stabilisers, and 0 is a stabiliser of every subset of V. In fact

Lemma 6.7: *The stabilisers of f form a subspace of V, and f is a union of cosets of this subspace.*

Proof: First we show that the stabilisers form a subspace. If v, w are distinct stabilisers of f then

$$f_{\langle v+w \rangle} = f + f_{v+w} = (f_v + f_w)_v = (f_v + f + f + f_w)_v = (f_{\langle v \rangle} + f_{\langle w \rangle})_v = (\emptyset + \emptyset)_v = \emptyset$$

so $v + w$ is also a stabiliser of f.

Now by definition if $x \in f$ and s is a stabiliser of f then $s + x \in f$, so f is a union of cosets of the stabiliser subspace. □

Definition: The *stabiliser dimension* of f is the dimension of the stabiliser subspace $\mathrm{Stab}(f)$ of f. □

The stabiliser dimension can sometimes give us information about the degree and form of f:

Lemma 6.8: *If f's stabiliser dimension is n (respectively $n-1$) then f is constant (a halfspace), and conversely.*

Proof: Suppose f has stabiliser dimension n. If f is not constant then there are points x and y such that $x \in f$, $y \notin f$. But every vector in V is a stabiliser of f, so in particular $x + y$ is a stabiliser, so $x \in f \implies x + (x + y) = y \in f$, a contradiction. Conversely it is clear that if f is constant then for all non-zero $v \in V$ we have $f = f_v \implies f_{\langle v \rangle} = \emptyset$, so every v is a stabiliser of f.

Now suppose that f has stabiliser dimension $n-1$. To show that f is a halfspace it is enough to show that the sum of any 3 vectors in f is also in f. So suppose x, y, z are in f. The 3 vectors $\{x + y, y + z, z + x\}$ span a 2-space, which must intersect the $(n-1)$-space $\mathrm{Stab}(f)$ in at least a 1-space. So without loss of generality $x + y \in \mathrm{Stab}(f)$. But now $z \in f \implies z + (x + y) = x + y + z \in f$ as claimed. Conversely by Lemma 1.17 $x^\perp$ is exactly its own stabiliser space, so it is also the stabiliser space of $x^{\bar{\perp}}$. □

Proposition 6.9: *A bent function f has stabiliser dimension 0.*

Proof: Suppose not — then there is some non-zero vector $v \in \mathrm{Stab}(f)$. Pick a vector x such that $x.v = 1$. Now translation by v takes points in $x^\perp$ to points in $x^{\bar{\perp}}$, and *vice versa*. But we know that translation by v fixes f setwise. Thus we must have $|f \cap x^\perp| = |f \cap x^{\bar{\perp}}|$ and this contradicts Proposition 1.16. □

Now suppose we start with a function f which is not constant. Then its stabiliser dimension is $< n$ so we can pick a vector $v_1 \notin \mathrm{Stab}(f)$ and calculate $f_{\langle v_1 \rangle}$. This is either constant or not, and it has lower degree than f. If it is not constant then we can pick $v_2 \notin \mathrm{Stab}(f_{\langle v_1 \rangle})$ and repeat the process to get $(f_{\langle v_1 \rangle})_{\langle v_2 \rangle} = f_{\langle v_1, v_2 \rangle}$.

Since the degree of each function is less than the degree of its predecessor, if we iterate this process it must terminate with a constant function $f_{\langle v_1, \ldots, v_l \rangle}$. No function in the sequence can be $\mathbb{O}$ because of the way we pick the v_is, so this last function must be $\mathbb{1}$. We note some facts about its predecessor, which will be useful later:

Proposition 6.10: *If $A \subseteq V$ and $v \neq 0$ is such that $A_{\langle v \rangle} = \mathbb{1}$ then*
(i) $|A| = 2^{n-1}$.

(ii) *The set of such* v*s,* $\{v : A_{\langle v\rangle} = \mathbb{1}\}$*, is a coset of* $\mathrm{Stab}(A)$.

Proof:

(i) A consist of exactly one point in each coset of $\langle v\rangle$.

(ii) If v_1, v_2 are non-zero then $A_{\langle v_1\rangle} = \mathbb{1} = A_{\langle v_2\rangle} \implies A_{v_1} = A_{v_2} \implies A_{v_1+v_2} = A$. □

We also have the following:

Lemma 6.11: *If* $U = \langle v_1, \ldots, v_l\rangle$ *then* $f_U = \mathbb{1}$ *iff every coset of* U *has odd intersection with* f.

Proof: $f_U(x) = 1$ iff x is in oddly many f_us with $u \in U$ iff oddly many $(x+u)$s are in f iff $|(x+U) \cap f|$ is odd. □

So we make the following:

Definition: If $f_U = \mathbb{1}$ then U is called an odd-space for f. □

So we have shown that any vector $v_1 \notin \mathrm{Stab}(f)$ is in some odd-space for f. Note that if f is bent this means that every non-zero vector is in an odd-space for f.

Now suppose that a bent function f has some odd-space $U = \langle u, v\rangle$ of dimension 2. Then

$$f_U = \mathbb{1} \implies f + f_u + f_v + f_{u+v} = \mathbb{1}.$$

All these translates of f are bent (Proposition 3.2), so if we can find a 2D odd-space $U = \langle u, v\rangle$ for f then we have found 4 bent functions summing to $\mathbb{1}$ and hence (by complementing one of them) 4 bent functions summing to $\mathbb{O}$, exactly as required by Corollary 5.5. Thus we obtain the bent function

$$F = (x_{n+1} + f_{\langle u\rangle} + 1)(x_{n+2} + f_{\langle v\rangle} + 1) + f$$

on $V(n+2, 2)$. Note that f can have several different 2D odd-spaces, yielding different bent functions.

This is not immediately much help, since given a bent function f the task of checking all the 2D subspaces of V in the hope that one of them is an odd-space for f is very slow. Even concentrating on those v_1s which produce $f_{\langle v_1\rangle}$ of low degree in the hope that they are in 2D odd-spaces does not help very much.

However, suppose that f was itself constructed using Corollary 5.5. Then $S = \langle e_{n-1}, e_n\rangle$ is a 2D odd-space for f. To see this, note that every row of f's matrix is bent, hence has odd weight, and these rows are precisely the cosets of S, which is therefore an odd-space.

Thus if we can form a bent function of n variables using Corollary 5.5, we can go on to form bent functions of $n+2, n+4, \ldots$ variables as well. For example, in Chapter 5 we formed the bent function

$$f = x_1x_2x_5 + x_1x_3x_6 + x_1x_4 + x_2x_3 + x_2x_5 + x_5x_6$$

in this way. We can check that $\langle e_5, e_6\rangle$ is an odd-space:

$$\begin{aligned}
f_{e_5} &= x_1x_2(x_5+1) + x_1x_3x_6 + x_1x_4 + x_2x_3 + x_2(x_5+1) + (x_5+1)x_6 \\
&= x_1x_2x_5 + x_1x_2 + x_1x_3x_6 + x_1x_4 + x_2x_3 + x_2x_5 + x_2 + x_5x_6 + x_6 \\
\implies f_{\langle e_5\rangle} &= (x_1x_2x_5 + x_1x_3x_6 + x_1x_4 + x_2x_3 + x_2x_5 + x_5x_6) + (x_1x_2x_5 \\
&\qquad + x_1x_2 + x_1x_3x_6 + x_1x_4 + x_2x_3 + x_2x_5 + x_2 + x_5x_6 + x_6) \\
&= x_1x_2 + x_2 + x_6 \\
\implies (f_{\langle e_5\rangle})_{e_6} &= x_1x_2 + x_2 + (x_6+1) \\
\implies f_{\langle e_5,e_6\rangle} &= (x_1x_2 + x_2 + x_6) + (x_1x_2 + x_2 + x_6 + 1) = 1.
\end{aligned}$$

So we calculate $f_{\langle e_6\rangle} = x_1x_3 + x_5$ in similar fashion and hence obtain the bent function

$$\begin{aligned}
F &= (x_7 + f_{\langle e_5\rangle} + 1)(x_8 + f_{\langle e_6\rangle} + 1) + f \\
&= (x_7 + x_1x_2 + x_2 + x_6 + 1)(x_8 + x_1x_3 + x_5 + 1) \\
&\qquad + x_1x_2x_5 + x_1x_3x_6 + x_1x_4 + x_2x_3 + x_2x_5 + x_5x_6 \\
&= x_1x_2x_8 + x_1x_3x_7 + x_1x_2 + x_1x_3 + x_1x_4 + x_2x_3 + x_2x_8 + x_5x_7 + x_6x_8 + x_7x_8 \\
&\qquad + x_2 + x_5 + x_6 + x_7 + x_8 + 1.
\end{aligned}$$

Note that this is *not* a non-singular quadratic, the only type of bent function of 8 variables we have seen so far. Also note that now $F_{\langle e_7,e_8\rangle} = (x_1x_3+x_5+x_8+1)_{\langle e_8\rangle} = 1$ so we could continue the process.

Originally this approach was prompted by the idea that given a bent function f we can pick any non-zero vector v and get $|f_{\langle v\rangle}| = 2^{n-1}$, which is necessary for v to be in a 2D odd-space for f. Thus we might hope that a bent function would have many 2D odd-spaces, leading to lots of new bent functions by this method. However, as we shall see in Chapter 8, this need not be true — a bent function can have no 2D odd-spaces at all.

Method 4 With arbitrary R and S we can take any two bent functions — g on R and h on S, say — and define $a_{rs} = g(r) + h(s)$. Now the rth row of A is just $g(r) + h(s) = \text{constant} + h(s)$, so is a bent function on S. Similarly each column of A is bent. Dualising the rows we get

$$f(r \mid s) = g(r) + h^*(s).$$

Method 5 We can extend this idea further. From Proposition 3.2 we know that as well as adding a constant to a bent function, we can also add any function in RM_1 and still produce a bent function. Thus if we pick a function $x^{\bar{\perp}} \in \mathrm{RM}_1$ on R, another $y^{\bar{\perp}} \in \mathrm{RM}_1$ on S, and bent functions g_0, g_1 on R and h_0, h_1 on S, we can define

$$a_{rs} = \left\{ \begin{matrix} g_0(r) & \text{if } s \in y^{\perp} \\ g_1(r) & \text{if } s \in y^{\bar{\perp}} \end{matrix} \right\} + \left\{ \begin{matrix} h_0(s) & \text{if } r \in x^{\perp} \\ h_1(s) & \text{if } r \in x^{\bar{\perp}} \end{matrix} \right\} = g_{s.y}(r) + h_{r.x}(s).$$

Now as in Method 4 the rth row of A is just one of the h_is (according to the value of $r.x$) plus one of $\mathbb{O}, \mathbb{1}, y^{\perp}, y^{\bar{\perp}}$ (according to the values of $g_0(r)$ and $g_1(r)$). Thus in any case this row is bent. A similar argument applies to the columns of A.

More explicitly, using Proposition 4.7 we have

$g_0(r)$	0	1	0	1
$g_1(r)$	0	0	1	1
The rth row of A	$h_{r.x}(s)$	$h_{r.x}(s) + y^{\perp}$	$h_{r.x}(s) + y^{\bar{\perp}}$	$h_{r.x}(s) + \mathbb{1}$
The dual of this row	${h_{r.x}}^*(s)$	${h_{r.x}}^*(s+y) + \mathbb{1}$	${h_{r.x}}^*(s+y)$	${h_{r.x}}^*(s) + \mathbb{1}$

Thus we obtain the bent function

$$\begin{aligned} f(r \mid s) &= {h_{r.x}}^*(s)(g_0(r)+1)(g_1(r)+1) + ({h_{r.x}}^*(s+y)+1)g_0(r)(g_1(r)+1) \\ &\quad {h_{r.x}}^*(s+y)(g_0(r)+1)g_1(r) + ({h_{r.x}}^*(s)+1)g_0(r)g_1(r) \\ &= ({h_{r.x}}^*(s) + {h_{r.x}}^*(s+y))(g_0(r)+g_1(r)) + {h_{r.x}}^*(s) + g_0(r). \end{aligned}$$

For an example of this construction, take $n = 2$, $R = \langle e_1, e_2 \rangle$, $S = \langle e_3, e_4 \rangle$,

$$g_0 = x_1x_2, \qquad g_1 = x_1x_2 + x_2, \qquad h_0 = x_3x_4 + 1, \qquad h_1 = x_3x_4 + x_3$$

and $x = 11$, $y = \cdot 1$. Then

$$A = \begin{bmatrix} \text{Matrix} \\ \text{with } g_i\text{s} \\ \text{as columns} \end{bmatrix} + \begin{bmatrix} \text{Matrix} \\ \text{with } h_i\text{s} \\ \text{as rows} \end{bmatrix} = \begin{matrix} \cdot\cdot\cdot\cdot \\ \cdot\cdot\cdot\cdot \\ \cdot\cdot 11 \\ 11\cdot\cdot \end{matrix} + \begin{matrix} 111\cdot \\ \cdot 1\cdot\cdot \\ \cdot 1\cdot\cdot \\ 111\cdot \end{matrix} = \begin{matrix} 111\cdot \\ \cdot 1\cdot\cdot \\ \cdot 111 \\ \cdot\cdot 1\cdot \end{matrix} \xrightarrow{\text{dualise rows}} \begin{matrix} 111\cdot \\ \cdot\cdot 1\cdot \\ 1\cdot\cdot\cdot \\ \cdot 1\cdot\cdot \end{matrix}$$

which is the support of

$$f = x_1x_4 + x_2x_3 + x_2x_4 + x_3x_4 + x_1 + 1,$$

which is a non-singular quadratic, and hence bent as expected. Checking the explicit

form, we have

$$\begin{aligned}
h_0{}^*(s) &= x_3x_4 + 1 \qquad \text{and} \qquad h_1{}^*(s) = x_3x_4 + x_4 \\
\implies h_{r.x}{}^*(s) &= (x_3x_4 + 1)(x_1 + x_2 + 1) + (x_3x_4 + x_4)(x_1 + x_2) \\
&= x_1x_4 + x_2x_4 + x_3x_4 + x_1 + x_2 + 1 \\
\implies h_{r.x}{}^*(s + y) &= x_1(x_4 + 1) + x_2(x_4 + 1) + x_3(x_4 + 1) + x_1 + x_2 + 1 \\
\implies h_{r.x}{}^*(s) + h_{r.x}{}^*(s + y) &= x_1 + x_2 + x_3 \\
\implies f &= (x_1 + x_2 + x_3)(x_1x_2 + x_1x_2 + x_2) \\
&\qquad + (x_1x_4 + x_2x_4 + x_3x_4 + x_1 + x_2 + 1) + x_1x_2 \\
&= x_1x_4 + x_2x_3 + x_2x_4 + x_3x_4 + x_1 + 1
\end{aligned}$$

as expected.

Method 6 Although searching for suitable matrices directly is slow, we can speed the process up somewhat. We can try to form a suitable matrix by picking rows which are the supports of bent functions at random and then checking the columns.

If we had to select all the rows before checking the columns this would still be very slow. However, note that one of the conditions on the columns which we have to check involves checking $|a_{\bullet s} + e_{n_r}{}^{\bar{\perp}}|$. By Proposition 1.16 we could, equivalently, check $|a_{\bullet s} \cap e_{n_r}{}^{\bar{\perp}}|$ or $|a_{\bullet s} \cap e_{n_r}{}^{\perp}|$. Now $e_{n_r}{}^{\perp}$ consists of the first 2^{n_r-1} points of R, so we can perform this check after selecting only the first 2^{n_r-1} rows of the matrix.

Another way of looking at this is to say that once we have chosen the first $2^{n_r-1} - 1$ rows, if a 2^{n_r-1}th row is possible at all then its entries are completely determined by this condition, so we can just calculate them directly.

By the same argument, once we have chosen the first $2^{n_r-1} + 2^{n_r-2} - 1$ rows, the entries of the $2^{n_r-1} + 2^{n_r-2}$th row are completely determined by the conditions involving $e_{n_r-1}{}^{\bar{\perp}}$ and $(e_{n_r-1} + e_{n_r})^{\bar{\perp}}$, and so on.

For example, if $n = 8$ and $n_r = n_s = 4$ we have 896 choices for each row, since there are this many bent functions of 4 variables (see Chapter 3). The first 7 rows determine the 8th, then the first 11 rows determine the 12th and so on. However, for many choices of the first 7 rows there is no 8th row which will satisfy the $x_4{}^{\bar{\perp}}$ condition, so we adopt a "backtracking" approach. We pick 7 rows and check them — if we can calculate the 8th row we do so and then go on to pick rows 9, 10 and 11, while if not we go back for a new set of 7 rows. Once we have 11 rows we check them and either calculate the 12th row or go back for a new choice of rows 9–11. We repeat this process until we have a set of 16 rows satisfying all the required conditions, at which point we have a suitable matrix.

Performance of the various methods for $n = 8$

Methods 1, 2, 4 and 5 are all closed-form, in the sense that they do not require any searches, and we can use them to find a large number of bent functions of 8 variables. However, many of these turn out to be in the same equivalence classes. As a simple example if two functions g and g' are equivalent then taking $g_1 = \ldots = g_4 = g$ and $g_1 = \ldots = g_4 = g'$ in Method 1 will produce two equivalent functions — recall that $f = (x_{n-1} + 1)(x_n + 1) + g$ and so

$$\begin{aligned} g' = g_v &\implies f' = f_v \\ g' = g + l^? &\implies f' = f + l^? \\ g' = \alpha g &\implies f' = \left(\begin{array}{c|c} \alpha & \mathbb{O} \\ \hline \mathbb{O} & \mathbb{I}_2 \end{array}\right) f. \end{aligned}$$

Using Method 3 (involving 2D odd-spaces) produces a few more classes. However, it is Method 6 which is the most powerful. We can use a computer to implement the backtracking search with $n_r = n_s = 4$ as described above, and the process is actually fast enough to obtain results in a reasonable time. This allows us to find large numbers of bent functions of 8 variables.

These methods have been implemented as C programs by the author. So far over 100 classes have been found, mostly using Method 6 — see Appendix A. In Chapter 8 we will see how we can show that these classes are indeed distinct.

7. Comments on the direct-summand construction

"Jordan-Hölder" conjectures

In Chapter 6 we saw how a bent function on a vector space V could be built up using bent functions on R and S, direct summands of V. However, since the only method that we have of constructing bent functions is this direct summand method, we cannot tell whether every bent function can be constructed in this way — for large n there seems to be no reason why this should be true.

Indeed, as we shall see in Chapter 8, there are certainly bent functions of 8 variables that do not have a 2D odd-space, and which can therefore be constructed by the method of Theorem 6.1 only if we take $n_r = n_s = 4$.

On the other hand, some bent functions can be constructed using more than one choice of R and S. For a simple example, consider $f = x_1x_2 + x_3x_4 + x_5x_6$. This has many 2D odd-spaces — in fact *every* vector is contained in some 2D odd-space. In particular $\langle e_1, e_2\rangle$ and $\langle e_3, e_4\rangle$ are both odd-spaces. Thus we can use either

$$R = \langle e_1, e_2\rangle\,,\ S = \langle e_3, e_4, e_5, e_6\rangle \qquad \text{or} \qquad R = \langle e_3, e_4\rangle\,,\ S = \langle e_1, e_2, e_5, e_6\rangle$$

to construct f using Theorem 6.1.

We might hope that if we can construct f on V from two direct sums in this way, we might be able to combine the constructions — in this example we would hope for a construction involving only bent functions on $\langle e_1, e_2\rangle$, $\langle e_3, e_4\rangle$ and $\langle e_5, e_6\rangle$. Slightly more generally, if f is a bent function on

$$V = A \oplus B \oplus C \oplus D$$

and f can be constructed using Theorem 6.1 with either

$$R = A \oplus B,\ S = C \oplus D \qquad \text{or} \qquad R = A \oplus C,\ S = B \oplus D,$$

we might hope for a construction involving bent functions on A, B, C, D. Such a construction would imply a result analogous to the Jordan-Hölder theorems for finite groups — we could "refine" factorisations of V to provide a construction of f in terms of bent functions on some set of minimal subspaces of V.

If this is going to work we would like the following two statements to be true:

Statement 7.1: *Consider a function f on*

$$V = (A \oplus B) \oplus (C \oplus D) = (A \oplus C) \oplus (B \oplus D)$$

where $\dim A, \ldots, \dim D$ *are all even. Let* f *be represented by* M *on* $(A \oplus B) \oplus (C \oplus D)$ *and by* N *on* $(A \oplus C) \oplus (B \oplus D)$*. Suppose that all the rows and columns of* M *and* N *are bent functions on the appropriate spaces (i.e.* M *and* N *are suitable matrices for Theorem 6.1). Then the restrictions of* f *to the cosets of* A *are bent functions.* □

Statement 7.2: *Consider a function* f *on*

$$V = (A \oplus B) \oplus (C \oplus D) = (A \oplus C) \oplus (B \oplus D)$$

where $\dim A, \ldots, \dim D$ *are all even, and suppose* f *and its restrictions to* $A \oplus B$ *and* $A \oplus C$ *are bent (so these two direct sums can each be used in Theorem 6.2). Then the restrictions of* f *to the cosets of* A *are bent functions.* □

If either of these were true we could try to use the cosets of A as the rows of a matrix, or perhaps a higher-dimensional structure, in some construction for f involving a direct sum with A as one of the summands.

However, in fact both these statements are false. In fact for $n = 8$

$$A = \langle e_1, e_2 \rangle\,,\ B = \langle e_3, e_4 \rangle\,,\ C = \langle e_5, e_6 \rangle\,,\ D = \langle e_7, e_8 \rangle \text{ and}$$
$$f = x_1x_3 + x_1x_5 + x_2x_4 + x_2x_6 + x_3x_4 + x_3x_6 + x_5x_6 + x_7x_8$$

provides a counterexample to *both* statements.

To check this, we observe that f is bent and calculate

$$M = \begin{array}{c} \cdots1\cdots1\cdots1111\cdot \\ \cdot1\cdots1\cdots1\cdot\cdot1\cdot11 \\ \cdot\cdot1\cdots1\cdots1\cdot11\cdot1 \\ \cdot111\cdot111\cdot1111\cdots \\ \cdot\cdot1\cdots1\cdots1\cdot11\cdot1 \\ 1\cdots1\cdots1\cdots\cdot111 \\ \cdots1\cdots1\cdots1111\cdot \\ 1\cdot111\cdot111\cdot11\cdot1\cdot\cdot \\ \cdots1\cdots1\cdots1111\cdot \\ \cdot1\cdots1\cdots1\cdot\cdot1\cdot11 \\ 11\cdot111\cdot111\cdot1\cdot\cdot1\cdot \\ 1\cdots1\cdots1\cdots\cdot111 \\ 11\cdot111\cdot111\cdot1\cdot\cdot1\cdot \\ \cdot111\cdot111\cdot1111\cdots \\ \cdots1\cdots1\cdots1111\cdot \\ 1\cdot111\cdot111\cdot11\cdot1\cdot\cdot \end{array} \quad \text{and} \quad N = \begin{array}{c} \cdots1\cdots1\cdots1111\cdot \\ \cdot1\cdots1\cdots1\cdot\cdot1\cdot11 \\ \cdot\cdot1\cdots1\cdots1\cdot11\cdot1 \\ \cdot111\cdot111\cdot1111\cdots \\ \cdots1\cdots1\cdots1111\cdot \\ 1\cdot111\cdot111\cdot11\cdot1\cdot\cdot \\ \cdot\cdot1\cdots1\cdots1\cdot11\cdot1 \\ 1\cdots1\cdots1\cdots\cdot111 \\ \cdot1\cdots1\cdots1\cdot\cdot1\cdot11 \\ \cdots1\cdots1\cdots1111\cdot \\ 1\cdots1\cdots1\cdots\cdot111 \\ 11\cdot111\cdot111\cdot1\cdot\cdot1\cdot \\ 1\cdot111\cdot111\cdot11\cdot1\cdot\cdot \\ \cdots1\cdots1\cdots1111\cdot \\ \cdot111\cdot111\cdot1111\cdots \\ 11\cdot111\cdot111\cdot1\cdot\cdot1\cdot \end{array}$$

Now all the rows and columns of these matrices are bent functions, so we have checked all the hypotheses of both statements. However, the restriction of f to A is $\mathbb{O}$, which is not bent, so neither statement is true.

Thus it seems that different ways of constructing f using Theorem 6.1 behave rather independently. In fact, it seems that being able to construct f in this way at all is something of an accident, which perhaps suggests that the majority of bent functions cannot be constructed in this way at all. Thus perhaps even our Method 6 can produce only a small proportion of bent functions.

Estimating the number of bent functions — upper bounds

In order to evaluate the success of our constructions, and estimate the completeness of Appendix A, we would like some estimate of the total number of bent functions, and the number of equivalence classes, for given $n = 2m$. Unfortunately the only known bounds on these numbers are very poor.

An upper bound comes from the fact that, by Proposition 1.14, a bent polynomial f has degree at most m. A polynomial of degree at most m is determined by the subset of the set of monomials of degree at most m which it contains. Since there are n variables $x_1, \ldots, x_n$, and each can appear in each monomial at most once, the number of possible monomials of degree d is just $\binom{n}{d}$, so the number of polynomials of degree at most m is 2 to the power $\sum_{i=0}^{m} \binom{n}{i}$.

So, for example,

$$\begin{array}{lll} n = 4 & \Longrightarrow & \text{bound} = 2^{1+4+6} = 2^{11} \\ n = 6 & \Longrightarrow & \text{bound} = 2^{1+6+15+20} = 2^{42} \\ n = 8 & \Longrightarrow & \text{bound} = 2^{1+8+28+56+70} = 2^{163} \end{array}$$

It is normally simplest to obtain bounds in terms of powers of 2. However, as a rough guide $2^{42} \approx 4.10^{12}$ and $2^{163} \approx 10^{49}$.

This upper bound is the one used by Preneel *et al.* [34], and is one of the simplest known, but also one of the best. We can improve it a little — for example we noted in Chapter 3 that there must be at least one monomial of degree $\geqslant 2$ so we do not have a completely free choice of the subset of monomials — but not really significantly.

Another bound is provided by the fact that we must have $|f| = 2^{n-1} \pm 2^{m-1}$. The number of functions of each of these weights is just $\binom{|V|}{|f|} = \binom{2^n}{2^{n-1}-2^{m-1}}$, so the number of bent functions is bounded by twice this number. However, this bound does not seem as good as the degree-based one:

$$\begin{array}{lll} n = 4 & \Longrightarrow & \text{bound} = 2\binom{16}{6} \approx 2^{14} \\ n = 6 & \Longrightarrow & \text{bound} = 2\binom{64}{28} \approx 2^{61} \\ n = 8 & \Longrightarrow & \text{bound} = 2\binom{256}{120} \approx 2^{252} \end{array}$$

As a special case, we can bound the number of bent functions with a given 2D odd-space. For example, in selecting a suitable $2^{n-2} \times 4$ matrix for Theorem 5.6

there are only 8 choices for each row, hence at most $8^{2^{n-2}}$ possible matrices. In the $n = 8$ case this bound is 2^{192}.

Again we can improve this bound, for example by using the fact that the weight of each column must be $2^{n-1} \pm 2^{m-1}$. This places 4 conditions on the frequencies of the 8 types of row, giving us a standard linear programming problem involving a simplex of solutions (see Greig [11], for example). Estimating the volume of this simplex in various ways then provides us with a bound on the number of functions, which for $n = 8$ turns out to be about 2^{187}.

We can extend this technique further by considering one or more of the other bent function conditions and using more variables to keep track of the number of rows of each type in various parts of the matrix. This gives us another, larger, linear programming problem and again we can estimate the volume of the solution simplex. For $n = 8$ we obtain about 2^{147}.

Given a bound on the number of bent functions with a *given* 2D odd-space, we can then multiply by the number of 2D subspaces of V, which is $(2^n - 1)(2^n - 2)/6$, to get a bound on the number of bent functions with *some* 2D odd-space.

However, this whole method does not help very much. In the $n = 8$ case the best final bound we can feasibly obtain is about 2^{160}, which is only just below the degree-based bound found above. Since this is bounding only a special subset of the bent functions, and a subset which we suspect to be proportionately rather small at that, this approach does not seem at all promising.

Estimating the number of bent functions — lower bounds

The known lower bounds on the number of bent functions for given n are also very poor. Essentially they consist of bounding the number of functions produced by a given construction.

For example, we saw in Chapter 4 that Maiorana's construction produces $2^{2^m} 2^m!$ bent functions. For small n this number is

$$\begin{array}{lcll} n = 4 & \Longrightarrow & \text{bound} = 2^4 4! & = 384 \\ n = 6 & \Longrightarrow & \text{bound} = 2^8 8! & \approx 2^{23} \\ n = 8 & \Longrightarrow & \text{bound} = 2^{16} 16! & \approx 2^{60} \end{array}$$

We can obtain other functions by applying the various equivalences of Chapter 3 to these. However, it seems hard to estimate the extent to which different functions produced in this way will actually be the same, rather than merely equivalent, making it hard to improve this bound. Even given a single function we shall see later that finding the size of its stabiliser — those equivalences which leave it unchanged — is rather hard, hence so is finding the size of its equivalence class.

One can try to find other lower bounds by considering the number of bent functions arising from the methods of Chapter 6 instead. However in practice for Methods 1–5 the bounds obtained are very poor since there are too few choices to make when performing the construction, while on the other hand Method 6 involves so much choice that bounding its performance is difficult.

Estimating the number of equivalence classes

Although finding the stabiliser of a given bent function is hard, we shall see that a randomly chosen bent function seems likely to have rather small, even trivial, stabiliser. Equivalently a random class is likely to have the same size as the group of equivalences $\mathrm{GB}(V)$.

Thus if we divide our upper bound on the number of functions by $|\mathrm{GB}(V)|$ we obtain an estimate of, although not strictly an upper bound on, the number of equivalence classes. For $n \leqslant 6$ this estimate is actually $\ll 1$, but for $n = 8$ it is about $2^{84} \approx 10^{25}$. So either our upper bound is much too high, or there are far too many classes for us ever to find a significant proportion of them explicitly, or both.

The following summary of these various estimates extends that of Preneel *et al.* [34]:

n	♯ classes	Maximal class size	♯ bent functions			♯ Boolean functions
			Lower bound	Actual	Upper bound	
2	1	192	8	8	8	16
4	1	$2^{23.3}$	384	896	2048	65536
6	4	$2^{47.2}$	$2^{23.3}$	$2^{32.3}$	2^{42}	2^{64}
8	$\geqslant 112$	$2^{79.2}$	$2^{60.3}$	$\geqslant 2^{80.8}$	2^{163}	2^{256}

8. Equivalence class invariants

A recurrent problem throughout our study of bent functions is that of determining whether two bent functions are equivalent under the action of $\mathrm{GB}(V)$.

An obvious way to approach this problem is by finding an equivalence class invariant, i.e. some function on the set of bent functions which is constant on each equivalence class. Normally an invariant can take the same value on different classes, so it can tell us that two bent functions are inequivalent, but not that they are equivalent.

We have already seen, and implicitly used, a simple invariant — the degree of f as a co-ordinate polynomial. As with any claimed invariant, we have to check that the three classes of operations generating $\mathrm{GB}(V)$ preserve the value of the invariant.

Clearly addition of an RM_1 function preserves the degree of a bent function, since we saw in Chapter 3 that this degree must be at least 2. We saw in the proof of Proposition 6.4 that translation also preserves degree. Finally consider applying a linear automorphism α to f. We saw in Chapter 3 this this is equivalent to replacing the variables of the polynomial by linear combinations of the variables. Since these are *linear* combinations any multiplying out produces terms of at most the original degree, so $\deg(\alpha f) \leqslant \deg(f)$. On the other hand α is invertible and α^{-1} is a linear automorphism, so $\deg(f) = \deg(\alpha^{-1}\alpha f) \leqslant \deg(\alpha f)$. Hence we must have equality here.

A useful series of invariants comes from consideration of odd-spaces and stabilisers, as in Chapter 6. Recall that the stabiliser of a subset $A \subseteq V$, $\mathrm{Stab}(A)$, is the subspace of vectors v such that $A \triangle A_v = \emptyset$.

Lemma 8.1: *If $A \subseteq V$ and $0 \neq v \in V$ then*

(i) $\overline{A}_v = \overline{(A_v)}$.

(ii) $\overline{A}_{\langle v \rangle} = A_{\langle v \rangle}$.

(iii) $\mathrm{Stab}\left(\overline{A}\right) = \mathrm{Stab}(A)$.

(iv) $\mathrm{Stab}(A_v) = \mathrm{Stab}(A)$.

Proof:

(i) $x \in \overline{A}_v \iff x + v \in \overline{A} \iff x + v \notin A \iff x \notin A_v \iff x \in \overline{(A_v)}$.

(ii) $\overline{A}_{\langle v \rangle} = \overline{A} \triangle \overline{A}_{\langle v \rangle} = \overline{A} \triangle \overline{(A_v)} = A \triangle A_v = A_{\langle v \rangle}$.

(iii) If $x \neq 0$ then $x \in LHS \iff \overline{A}_{\langle x \rangle} = \emptyset \iff A_{\langle x \rangle} = \emptyset \iff x \in RHS$.

(iv) If $x \neq 0$ then $x \in LHS \iff (A_v)_{\langle x \rangle} = \emptyset \iff A_{v+\langle x \rangle} = \emptyset \iff A_{\langle x \rangle} = \emptyset_v = \emptyset \iff x \in RHS$. □

We will mainly be considering functions of the form $A_{\langle v\rangle}$, so we need to know how the stabilisers of these behave under the action of the various equivalences:

Lemma 8.2: *If $A \subseteq V$, $v, x \in V$ and $\alpha \in \mathrm{GL}(V)$ then*

(i) $\mathrm{Stab}(\overline{A}_{\langle v\rangle}) = \mathrm{Stab}(A_{\langle v\rangle})$.

(ii) $\mathrm{Stab}((A \triangle x^{\bar{\perp}})_{\langle v\rangle}) = \mathrm{Stab}(A_{\langle v\rangle})$.

(iii) $\mathrm{Stab}((A_x)_{\langle v\rangle}) = \mathrm{Stab}(A_{\langle v\rangle})$.

(iv) $\mathrm{Stab}((\alpha A)_{\langle \alpha v\rangle}) = \alpha\,\mathrm{Stab}(A_{\langle v\rangle})$.

Proof:

(i) is immediate by Lemma 8.1(ii).

(ii) $(A \triangle x^{\bar{\perp}})_{\langle v\rangle} = A_{\langle v\rangle} \triangle x^{\bar{\perp}}{}_{\langle v\rangle} = A_{\langle v\rangle}$ up to complementation, by Lemma 1.17. Hence, using (i) if necessary, we have the result.

(iii) $(A_x)_{\langle v\rangle} = A_{x+\langle v\rangle} = (A_{\langle v\rangle})_x$ so we have the result by Lemma 8.1(iv).

(iv) If $s \neq 0$ then

$$\begin{aligned}
s \in \mathrm{Stab}((\alpha A)_{\langle \alpha v\rangle}) &\iff ((\alpha A)_{\langle \alpha v\rangle})_{\langle s\rangle} = \emptyset \\
&\iff \alpha A \triangle (\alpha A)_{\alpha v} \triangle (\alpha A)_s \triangle (\alpha A)_{\alpha v+s} = \emptyset \\
&\iff \alpha A \triangle \alpha(A_v) \triangle \alpha(A_{\alpha^{-1}s}) \triangle \alpha(A_{v+\alpha^{-1}s}) = \emptyset \\
&\iff \alpha(A \triangle A_v \triangle A_{\alpha^{-1}s} \triangle A_{v+\alpha^{-1}s}) = \emptyset \\
&\iff A \triangle A_v \triangle A_{\alpha^{-1}s} \triangle A_{v+\alpha^{-1}s} = \alpha^{-1}\emptyset = \emptyset \\
&\iff (A_{\langle v\rangle})_{\langle \alpha^{-1}s\rangle} = \emptyset \\
&\iff \alpha^{-1}s \in \mathrm{Stab}(A_v) \\
&\iff s \in \alpha\,\mathrm{Stab}(A_v).
\end{aligned}$$

□

So if we consider the set of stabiliser spaces $\{\mathrm{Stab}(A_{\langle v\rangle}) : v \in V\}$ this lemma shows that the first 3 equivalences leave them unchanged, while applying a linear automorphism to A takes them to another set with the same dimensions. This suggests the following:

Definition: If $A \subseteq V$ then we partition V into sets $D_0, D_1, \ldots, D_n$, where

$$D_i(A) = \{v \in V \setminus \{0\} : \dim(\mathrm{Stab}(A_{\langle v\rangle})) = i\}.$$

Then let $d_i(A) = |D_i(A)|$, and denote the set of $d_i(A)$s by $\underline{d}(A)$. □

So Lemma 8.2 immediately implies

Proposition 8.3: *If $A \subseteq V$, $x \in V$ and $\alpha \in \mathrm{GL}(V)$ then for $0 \leqslant i \leqslant n$*

(i) $D_i\left(\overline{A}\right) = D_i(A)$.

(ii) $D_i(A \triangle x^{\bar{\perp}}) = D_i(A)$.
(iii) $D_i(A_x) = D_i(A)$.
(iv) $D_i(\alpha A) = \alpha D_i(A)$. □

Since in particular we can take A to be the support of a bent function f, we see that

Theorem 8.4: *$\underline{d}(\bullet)$ is a bent function equivalence class invariant.* □

Before we actually use $\underline{d}(\bullet)$ as an invariant, what else can we say about it? Clearly $d_0, \ldots, d_n$ must sum to $|V| - 1 = 2^n - 1$. In fact d_0 must be zero, since the stabiliser of $A_{\langle v \rangle}$ certainly contains v, so cannot have dimension 0.

At the other extreme, the only sets for which every vector is a stabiliser are $\emptyset$ and V, and if $A_{\langle v \rangle}$ is one of these we must have $|A| = 2^{n-1}$. Hence if f is a bent function (so that $|f| = 2^{n-1} \pm 2^{m-1}$) then $d_n(f) = 0$.

We can also say quite a lot about $D_{n-1}(A)$ and $d_{n-1}(A)$:

Proposition 8.5: *If* $\mathrm{Stab}(A_{\langle v \rangle}) = a^\perp$ *and* $\mathrm{Stab}(A_{\langle w \rangle}) = b^\perp$, *where v, w are distinct and non-zero, then* $\mathrm{Stab}(A_{\langle v+w \rangle}) = (a+b)^\perp$ *(unless* $|A| = 2^{n-1}$ *and* $A_{\langle v+w \rangle} = \emptyset$ *or* V*).*

Proof: $\mathrm{Stab}(A_{\langle v \rangle})$ has dimension at least $n-1$ so by Lemma 6.8 $f_{\langle v \rangle}$ is a halfspace and so is taken either to itself or to its complement by translations, by Lemma 1.17. Similarly for $A_{\langle w \rangle}$.

So pick $s \in (a+b)^\perp$. Then $s \in a^\perp \iff s \in b^\perp$ so translation by s either fixes both of $A_{\langle v \rangle}$, $A_{\langle w \rangle}$ or takes each to its complement. In either case

$$\begin{aligned}
& (A_{\langle v \rangle})_s + (A_{\langle w \rangle})_s = A_{\langle v \rangle} + A_{\langle w \rangle} \\
\implies & A_s + A_{v+s} + A_s + A_{w+s} = A + A_v + A + A_w \\
\implies & A_{v+s} + A_{w+s} = A_v + A_w \\
\implies & A_s + A_{v+w+s} = A + A_{v+w} \qquad \text{translating by } v \\
\implies & (A_{\langle v+w \rangle})_s = A_{\langle v+w \rangle}
\end{aligned}$$

so $s \in \mathrm{Stab}(A_{\langle v+w \rangle})$. Thus we have shown that $(a+b)^\perp \subseteq \mathrm{Stab}(A_{\langle v+w \rangle})$, and we must have equality except where stated. □

Proposition 8.6: *$D_{n-1}(f) \cup \{0\}$ is a subspace T of V, and each $D_i(f)$ is a union of cosets of it.*

Proof: We prove both parts at once: if $v \neq w$ with $v \in D_i(f)$ and $w \in D_{n-1}(f)$ then $f + f_w$ is some linear function $a^?$ and so

$$f_{\langle v+w \rangle} = f + f_{v+w} = (f_w + f_v)_w = (f + a^? + f_v)_w = (f + f_v)_w + a^?.$$

Thus by Lemma 8.2

$$\mathrm{Stab}(f_{\langle v+w\rangle}) = \mathrm{Stab}((f+f_v)_w + a^?) = \mathrm{Stab}(f+f_v)$$

which has dimension i by our choice of v, so $v+w \in D_i(f)$. □

Corollary 8.7: *For some $0 \leqslant t \leqslant n$ we have $d_{n-1}(A) = 2^t - 1$ and $2^t \mid d_i(A)$ for all $1 \leqslant i \leqslant n$.* □

For the proof of the next result it is convenient to introduce the symbol "$\doteq$" to mean "is equal to up to complementation".

Theorem 8.8: *There is a bijection between T and $T^* = D_{n-1}(f^*) \cup \{0\}$.*

Proof: If $0 \neq t \in T$ then $f + f_t = u^{(\sigma)}$ for some u, σ, and using Proposition 4.7

$$f \doteq f_t + u^{\bar{\perp}} \implies f^* \doteq \left(f^* + t^{\bar{\perp}}\right)_u \implies f^* + (f^*)_u = t^? \implies u \in T^*.$$

So we can define a map $T \to T^*$ by $t \mapsto u$, and since $f^{**} = f$ this is a bijection. □

Corollary 8.9: $d_{n-1}(f) = d_{n-1}(f^*)$. □

Finally we consider $d_{n-2}(A)$:

Lemma 8.10: *If $A \subseteq V$ with $|A| = 2^{n-1}$ then* $\dim(\mathrm{Stab}(A)) \neq n-2$.

Proof: Let S=Stab(A), and suppose $\dim(S) = n-2$, so $|S| = 2^{n-2}$. A is a disjoint union of cosets of S, but it has weight 2^{n-1}, so it is the union of exactly two cosets. Hence it is a halfspace (after factoring out S, A is a function of 2 variables of weight 2 — all such are halfspaces). But then its stabiliser has dimension $n-1$, a contradiction. □

Now if f is bent Theorem 1.19 says that $|f_{\langle v\rangle}| = 2^{n-1}$ for every non-zero $v \in V$, so we must have $d_{n-2}(f) = 0$.

We summarise these results about $\underline{d}(\bullet)$ as follows:

Theorem 8.11: *If f is a bent function on V then for some $0 \leqslant t \leqslant n$ we have $d_{n-1}(f) = d_{n-1}(f^*) = 2^t - 1$, $2^t \mid d_i(f)$ and $2^t \mid d_i(f^*)$ for $1 \leqslant i \leqslant n-3$, and every other $d_i(f)$ and $d_i(f^*)$ is 0.* □

We will thus sometimes write $\underline{d}(f)$ as $[d_1, \ldots, d_{n-1}]$ — the penultimate entry d_{n-2} will always be 0.

Given f we can calculate $\underline{d}(f)$ easily by computer — calculations performed using the author's C programs are summarised in Appendix B. $\underline{d}(\bullet)$ turns out to be surprisingly good at distinguishing equivalence classes. For example, it can distinguish the four classes for $n = 6$. Note that in many cases with $n = 8$, $\underline{d}(f) \neq \underline{d}(f^*)$,

so that f and f^* cannot be equivalent, whereas we saw in Chapter 4 that they *are* always equivalent for $n \leqslant 6$.

In fact, recall that Corollary 4.8 says that if f and g are equivalent then so are f^* and g^*. This means that even if $\underline{d}(f) = \underline{d}(g)$, provided that $\underline{d}(f^*) \neq \underline{d}(g^*)$ we can still deduce that f and g are inequivalent. This allows us to say that classes 8.77 and 8.79, for example, are distinct.

However, the fact that all the $n = 8$ classes have different values of $\underline{d}$ (when considered with their duals) is rather misleading. This is because in using the various methods of Chapter 6 to find the functions listed in Appendix A this invariant was the one used to distinguish equivalence classes, because it is fairly cheap to calculate. However for $n = 8$ the main method used, Method 6, tends to produce functions with a few common values of $\underline{d}$ — in fact the majority of them have $\underline{d} = [255, 0, \ldots, 0]$ as for class 8.84. Thus relying on this invariant means that many classes may have been discarded unnecessarily.

It is noticeable that most of the $\underline{d}$s seem to have larger entries at the beginning. This agrees with the above observation about the majority of functions produced by Method 6. While this may be a special feature of Method 6, it does seem reasonable as a general effect. Larger entries at the end of $\underline{d}$ indicate that f is "close to being symmetrical", whereas we would expect that a typical f would not be, so would have the observed form of $\underline{d}$.

Otherwise there does not seem to be much structure to the $\underline{d}$s beyond that guaranteed by Theorem 8.11. However, we can prove some results about the form of $\underline{d}$ for various special functions. For example, we can consider non-singular quadratics.

Theorem 8.12: *If f is a bent function then f is a non-singular quadratic iff $\underline{d}(f) = [0, \ldots, 0, 2^n - 1]$.*

Proof:

$\Rightarrow$: For any $v \in V$, by Proposition 6.4 $f_{\langle v \rangle}$ has lower degree than f. If f is a quadratic this means that every $f_{\langle v \rangle}$ is in RM_1. Hence by Lemma 6.8 every $f_{\langle v \rangle}$ has stabiliser dimension $n - 1$, so every $v \in D_{n-1}(f)$.

$\Leftarrow$: Suppose f is not a non-singular quadratic. Then f has degree $\geqslant 3$, so it contains a term T of maximal degree $d \geqslant 3$, which without loss of generality is $x_1 x_2 \ldots x_d$. Now let $v = e_1$ and consider f_v — this is obtained from f by replacing x_1 by $(x_1 + 1)$.

Multiplying out these brackets introduces an extra term $x_2 \ldots x_d$ from T — conversely T is the *only* term of f from which an extra $x_2 \ldots x_d$ can come. Thus $f + f_v$ contains $x_2 \ldots x_d$, so is not linear, so cannot have stabiliser dimension

$n-1$, by Lemma 6.8. Hence $v \notin D_{n-1}$ so $d_{n-1} < 2^n - 1$. □

We can also consider functions produced by Maiorana's construction (Proposition 4.2), but first we need to investigate the construction in more detail. If $V = A \oplus B$ and π is a permutation of the points of A then we want to write the function $f(x \mid y) = (\pi x).y$ as a polynomial in the co-ordinates of V.

For a fixed value of x the function f is just some sum of the co-ordinates of B, the co-ordinates involved depending on the image πx. So we need to find a polynomial which takes the value 1 on the singleton point x and multiply this by the appropriate co-ordinates of B. f is then the sum of the products obtained as x ranges over A.

As an example, consider $V = A \oplus B = \langle e_1, e_2 \rangle \oplus \langle e_3, e_4 \rangle$ and let π be the 3-cycle $(\cdot\cdot, \cdot 1, 11)$ as in Chapter 4. Then we have

x	πx	$(\pi x).y$	Singleton polynomial	Product
$\cdot\cdot$	$\cdot 1$	x_4	$(x_1+1)(x_2+1)$	$x_1x_2x_4 + x_1x_4 + x_2x_4 + x_4$
$1\cdot$	$1\cdot$	x_3	$x_1(x_2+1)$	$x_1x_2x_3 + x_1x_3$
$\cdot 1$	11	$x_3 + x_4$	$(x_1+1)x_2$	$x_1x_2x_3 + x_2x_3 + x_1x_2x_4 + x_2x_4$
11	$\cdot\cdot$	$\mathbb{O}$	x_1x_2	$\mathbb{O}$

so the required sum is $(x_1x_2x_4 + x_1x_4 + x_2x_4 + x_4) + (x_1x_2x_3 + x_1x_3) + (x_1x_2x_3 + x_2x_3 + x_1x_2x_4 + x_2x_4) = x_1x_3 + x_1x_4 + x_2x_3 + x_4$.

Theorem 8.13: *If f is essentially constructed by Maiorana's construction then*

$$d_m + d_{m+1} + \ldots + d_{n-1} \geqslant 2^m - 1.$$

Proof: By Theorem 8.4 we may assume that f really is constructed by Maiorana's construction. Write $V = A \oplus B = \langle e_1, \ldots, e_m \rangle \oplus \langle e_{m+1}, \ldots, e_n \rangle$ — f can be written as

$$f(x \mid y) = g(x) + (\pi x).y.$$

However, from the above discussion about writing f as a polynomial we see that each term of this polynomial contains exactly one of the variables $\{x_{m+1}, \ldots, x_n\}$. So pick a vector $b \in B$ and consider f_b. This is obtained from f by replacing some of the variables x_i by $(x_i + 1)$, but at most one such substitution is necessary per term.

Thus if we multiply out these brackets, the extra terms we have introduced contain only the variables $x_1, \ldots, x_m$. This means that $x_{m+1}, \ldots, x_n$ do not appear in $g = f + f_b$, so each of $e_{m+1}, \ldots, e_n$ is a stabiliser of g. Thus g has stabiliser dimension at least m.

Now since b was arbitrary in B we see that there are at least $2^m - 1$ non-zero vectors b such that $f_{\langle b \rangle}$ has stabiliser dimension $\geqslant m$, which is the result claimed. □

The last column of Appendix B indicates which classes cannot be produced by Maiorana's construction according to this criterion — it will be seen that the majority cannot.

Going in the opposite direction, we can use $\underline{d}$ to tell us facts about f, for example that f is quadratic (Theorem 8.12). As another example, it can (sometimes) tell us that f has a 2D odd-space:

Proposition 8.14:

(i) *If $A \subset V(3,2)$, $|A| = 2^{n-1}$ then A can be translated to its complement.*

(ii) *If $A \subset V(n,2)$ and $\dim(\mathrm{Stab}(A)) \geqslant n-3$ then A can be translated to its complement.*

(iii) *If f is a bent function and $d_{n-3} + d_{n-1} > 0$ then f has a 2D odd-space.*

Proof:

(i) Check by exhaustive search. This is made easier by noting that we need only consider one subset A from each equivalence class with respect to linear automorphisms, translation and complementation. However in $V(3,2)$ there are only two such classes of functions of weight 2^{n-1} — the class of all halfspaces, which can clearly be translated to their complements, and the class of all non-halfspaces, represented by $\{\cdots, 1\cdot\cdot, \cdot 1\cdot, \cdot\cdot 1\}$ which is translated to its complement by 111.

(ii) A is a disjoint union of cosets of $S = \mathrm{Stab}(A)$, so we can factor out S and apply (i).

(iii) We can pick $v \in D_{n-3} \sqcup D_{n-1}$ — then $f_{\langle v \rangle}$ can be translated to its complement by (ii). □

Note that the converse to (iii) is false — a function can fail to satisfy the condition, but still have a 2D odd-space. For example if

$$\begin{aligned} f = {} & x_2x_3x_4x_8 + x_5x_6x_7x_8 + x_1x_2x_8 + x_1x_3x_8 + x_1x_4x_5 + x_1x_4x_6 + x_1x_4x_8 \\ & + x_1x_7x_8 + x_2x_3x_8 + x_2x_4x_7 + x_2x_5x_8 + x_2x_6x_8 + x_3x_4x_7 + x_3x_4x_8 \\ & + x_5x_6x_7 + x_5x_6x_8 + x_5x_7x_8 + x_1x_4 + x_1x_7 + x_2x_5 + x_2x_6 + x_2x_7 \\ & + x_3x_5 + x_3x_7 + x_3x_8 + x_4x_6 + x_4x_8 + x_5x_6 + x_6x_7 + x_6x_8 + x_6, \end{aligned}$$

the representative of class 8.37, then $d_5 = d_7 = 0$ but $\langle e_1, e_2 + e_7 \rangle$ is an odd-space.

In fact functions with no 2D odd-space at all seem rather rare, at least among those constructed using the methods of Chapter 6. However, they do exist —

class 8.70 is an example. By the Method 3 discussion in Chapter 6 this means that this class cannot be constructed by the four-function construction (Corollary 5.5). This answers a question raised by Meier and Staffelbach [30].

The stabiliser graph

While $\underline{d}$ is quite effective as an invariant, it seems inefficient in that we are only counting the points in the D_is rather than using any information about their structure. We can encode some of this structure as follows:

Definition: If f is a bent function the *stabiliser graph* of f, $\Gamma(f)$, is a graph on the points of V with adjacency relation

$$x \sim y \iff f_{\langle x,y\rangle} = \mathbb{O}$$

for $x \neq y$. □

Clearly $\Gamma(f)$ is a simple undirected graph. Note that if $x, y \neq 0$ then

$$x \sim y \iff x \in \mathrm{Stab}(f_{\langle y\rangle})$$

which justifies our statement that Γ encodes information about the D_is — they are the sets of vertices of each degree.

Lemma 8.2 says that applying translations and adding RM_1 functions to f leave $\Gamma(f)$ invariant, and we obtain an isomorphism from $\Gamma(f)$ to $\Gamma(\alpha f)$ by applying α^{-1} to V. Thus although Γ is not quite an invariant of an equivalence class, the isomorphism class of Γ is. Hence we can use any graph-theoretic isomorphism invariant and apply it to Γ to get an equivalence class invariant for bent functions.

Although in general testing graph isomorphism is hard, there are various cheaper invariants that we can use. For example, $\underline{d}$ is clearly just the graph-theoretic degree sequence of Γ. Similarly we can consider the neighbours of the highest degree vertices of Γ, and find their degrees, and so on.

What happens if in examining the graphs of two bent functions we find that they are in fact isomorphic? While we cannot deduce that the functions are necessarily equivalent, we know that if α is a linear automorphism involved in an equivalence it induces a graph isomorphism. We can partition the graph's vertices into sets which are fixed setwise by the graph's automorphism group — the crudest attempt at this is just to use sets of vertices of each degree. We then know that α must take each set of vertices of the first graph to the corresponding set in the second graph. Since α must also be a linear automorphism of V this can be a powerful restriction on α. We shall see this method in action in Chapter 10.

Other invariants

Γ still seems to throwing away a lot of information about f, since it considers f_S only for 2-dimensional S. We could, however, define a similar k-ary relation $\mathcal{R}$ on $V^k = V \times V \times \ldots \times V$ for any $1 \leqslant k \leqslant n$, as follows:

$$\mathcal{R}(v_1, \ldots, v_k) = \begin{cases} 1 & \text{if } f_{\langle v_1, \ldots, v_k \rangle} = \mathbb{O} \\ 0 & \text{otherwise.} \end{cases}$$

We can extend this yet further to define a k-dimensional array A with each dimension indexed by the points of V

$$A_{v_1 \ldots v_k} = |f_{\langle v_1, \ldots, v_k \rangle}|.$$

As with Γ an equivalence of bent functions induces an isomorphism of their corresponding arrays — letting the linear automorphism part of the equivalence act on the indices of each dimension preserves the array.

In fact all these invariants are examples of a more general class of structures which are respected by bent function equivalences. We shall discuss this class in more detail in Chapter 12.

9. Designs

We use the usual definition of a design — see Cameron and van Lint [7], for example:

Definition: A t-(v, k, λ) *design* is a pair of sets $(\mathcal{P}, \mathcal{B})$ (the *points* and *blocks* of the design) and an incidence relation I on $\mathcal{P} \times \mathcal{B}$ such that

(i) $|\mathcal{P}| = v$.

(ii) Each block $B \in \mathcal{B}$ is incident with exactly k points in $\mathcal{P}$.

(iii) Any t points in $\mathcal{P}$ are mutually incident with exactly λ blocks in $\mathcal{B}$.

We normally insist that $\mathcal{P}$ and $\mathcal{B}$ are non-empty and that $v \geqslant k \geqslant t$ so $\lambda > 0$ to avoid trivial cases. □

A t-(v, k, λ) design is sometimes just called a t-design. If no two blocks are incident with the same set of points then the design is called *simple*. In this case we can identify a block with the set of points incident with it, and say that the block contains these points. All the designs we consider will be simple.

We note the following useful facts about t-designs — these are noted in [7] and proved by straightforward counting arguments:

Proposition 9.1: *If $\mathcal{D} = (\mathcal{P}, \mathcal{B})$ is a t-(v, k, λ) design with $|\mathcal{B}| = b$ then*

(i) *If $s \leqslant t$ then $\mathcal{D}$ is also an s-design.*

(ii) *If $t \geqslant 1$ then there is some fixed r such that each point is in exactly r blocks.*

(iii) $bk = vr$.

(iv) *If $t = 2$ then $r(k-1) = \lambda(v-1)$.* □

We can conveniently record the incidence relation of a design using a matrix:

Definition: If $\mathcal{D} = (\mathcal{P}, \mathcal{B})$ is a design then the *adjacency matrix* of $\mathcal{D}$ is a matrix A with rows indexed by the blocks and columns indexed by the points, such that

$$a_{BP} = \begin{cases} 1 & \text{if } P \in B \\ 0 & \text{otherwise.} \end{cases}$$

□

Note that although this basic idea is fairly standard, some authors use the transpose of this definition.

For an example of a 2-design let $\mathcal{P}$ be the non-zero points of some vector space $V = V(n, 2)$ with $n \geqslant 2$ and let $\mathcal{B}$ be the 2-spaces of V. Then every block contains exactly 3 points of $\mathcal{P}$, and any pair of points is contained in exactly one block. Thus we have a 2-$(2^n - 1, 3, 1)$ design.

If we take $n = 3$ here then the design has 7 points and 7 blocks, and is an example of the following special case:

Definition: If a design $\mathcal{D} = (\mathcal{P}, \mathcal{B})$ is such that $|\mathcal{P}| = |\mathcal{B}|$ ($v = b$) then $\mathcal{D}$ is called a *symmetric* or *square* design. □

This 2-$(7, 3, 1)$ design is the "paradigmatic example" of a square design (Lander [24]). In fact it is the unique design with these parameters, and we shall meet it several times in various guises.

The fact that a design is square tells us quite a lot more about its parameters and structure:

Proposition 9.2: *If $\mathcal{D}$ is a t-(v, k, λ) design then with the notation of Proposition 9.1 the following are equivalent:*

(i) *$v = b$ ($\mathcal{D}$ is square).*

(ii) *$r = k$.*

(iii) *Any two blocks contain λ common points.*

(iv) *Any two blocks contain a constant number of common points.*

Proof: Lander [24 Theorem 1.11] or Cameron and van Lint [7 Theorem 1.15]. □

Given a design $\mathcal{D}$ we can calculate its adjacency matrix A as above and hence find its transpose A^T. We can regard this as the adjacency matrix of another design, called the dual of $\mathcal{D}$, which we write as $\mathcal{D}^T$. Its point set corresponds to $\mathcal{D}$'s block set, and *vice versa*. In general all we can say about $\mathcal{D}^T$ is that it is a 1-design — this follows immediately from Proposition 9.1(ii). However:

Proposition 9.3: *If $\mathcal{D}$ is a square 2-(v, k, λ) design then so is $\mathcal{D}^T$.*

Proof: $\mathcal{D}^T$'s point set ($\mathcal{D}$'s block set) has size v by hypothesis. Each block is incident with k points by Proposition 9.1(ii). Each pair of points is in λ blocks by Proposition 9.2(iii). □

Inducing a design from a bent function

Given a bent function f consider the various functions $f + x^?$ as $x^?$ ranges over RM_1. By Proposition 1.10 these all have weight $2^{n-1} \pm 2^{m-1}$, but consider just those with the same weight as f, i.e. define

$$\mathcal{B} = \{f + x^? : x^? \in \mathrm{RM}_1 \text{ and } |f + x^?| = |f|\}.$$

Proposition 9.4: *If f is a light bent function then $\mathcal{A}(f) = (V, \mathcal{B})$ is a square 2-design with parameters*

$$2\text{-}(2^n, 2^{n-1} - 2^{m-1}, 2^{n-2} - 2^{m-1}).$$

Proof: Clearly $|V| = 2^n$ and by definition $|B| = 2^{n-1} - 2^{m-1}$ for each $B \in \mathcal{B}$, so v and k are as claimed.

It is also clear that $|f + x^{\perp}|$ and $|f + x^{\bar{\perp}}|$ have different weights, so exactly one of them is light, so there is a 1–1 correspondence between blocks $B \in \mathcal{B}$ and points $x \in V$. So if we have two distinct blocks $B_1, B_2 \in \mathcal{B}$ they correspond to distinct points $v_1, v_2 \in V$. Then

$$\begin{aligned} 2|B_1 \cap B_2| &= |B_1| + |B_2| - |B_1 \triangle B_2| \\ &= 2^{n-1} - 2^{m-1} + 2^{n-1} - 2^{m-1} - |f + {v_1}^? + f + {v_2}^?| \\ &= 2^n - 2^m - |{v_1}^? + {v_2}^?| = 2^n - 2^m - |(v_1 + v_2)^?|. \end{aligned}$$

Now if $v_1 \neq v_2$ then $(v_1 + v_2)^?$ is a halfspace, so has weight 2^{n-1}, so λ is as claimed.

The 1–1 correspondence also tells us that $|\mathcal{B}| = |V|$, i.e. that the design is square. □

The following can be proved in exactly the same way:

Proposition 9.5: *If f is a heavy bent function then $\mathcal{A}(f) = (V, \mathcal{B})$ is a square 2-design with parameters*

$$2\text{-}(2^n, 2^{n-1} + 2^{m-1}, 2^{n-2} + 2^{m-1}).$$

□

We call this design the *addition design* of f (hence the notation $\mathcal{A}(f)$). Note that in fact the blocks of $\mathcal{A}\left(\overline{f}\right)$ are just the complements of those of $\mathcal{A}(f)$, so usually we only consider light f — most of the results we prove can easily be converted for heavy f.

In the light of the 1–1 correspondence described in the proof of Proposition 9.4 we will sometimes talk of "the block b" (for some point $b \in V$) when we mean "the block $f + b^?$". As it stands, given $p, b \in V$ it seems hard to tell whether $p \in b$ (i.e. whether the point p is in the block $f + b^?$). We have to work out which of $f + b^{\perp}$ or $f + b^{\bar{\perp}}$ has the correct weight to be a block of the design and then test whether p is in the support of the appropriate function.

In such calculations it is often convenient to consider a statement such as "$a = 23$" or "$x \in f$" as an expression taking the value 1 if it is true and 0 if it is false — we sometimes enclose an expression in square brackets to emphasise this. This convention allows us to encapsulate the above calculation in the following identity:

Theorem 9.6: *If f is a bent function and $p, b \in V$ then in $\mathcal{A}(f)$*

$$p \in b = f(p) \;\hat{+}\; f^*(b) \;\hat{+}\; f^*(0) \;\hat{+}\; p.b.$$

Proof: First suppose that f and $f + b^{\bar{\perp}}$ have the same weight. Then

$$p \in b = p \in f + b^{\bar{\perp}} = p \in f \ \hat{+} \ p \in b^{\bar{\perp}} = f(p) \ \hat{+} \ p.b.$$

This accounts for the first and fourth terms of the result.

Now if f and $f + b^{\bar{\perp}}$ in fact have different weights this expression calculates exactly the wrong answer. Thus we must 1 to the expression iff f and $f + b^{\bar{\perp}}$ have different weights iff $f^*(0) \neq f^*(b)$ (by Proposition 4.5), so we must add $f^*(b) \ \hat{+} \ f^*(0)$. □

Corollary 9.7: *The blocks of $\mathcal{A}(f)$ are of the form $f + b^{(f^*(b) \ \hat{+} \ f^*(0) \ \hat{+} \ 1)}$ for $b \in V$.* □

Corollary 9.8: *The 0th block of $\mathcal{A}(f)$ is f.* □

This form of the incidence relation allows us to demonstrate a neat correspondence between bent function dualities and those of their designs:

Theorem 9.9: *If f is a bent function then $\mathcal{A}(f^*) = (\mathcal{A}(f))^T$.*

Proof:

$$\begin{aligned} p \in b \text{ in } \mathcal{A}(f^*) &= f^*(p) \ \hat{+} \ f^{**}(b) \ \hat{+} \ f^{**}(0) \ \hat{+} \ p.b \\ &= f^*(p) \ \hat{+} \ f(b) \ \hat{+} \ f(0) \ \hat{+} \ b.p \\ &= b \in p \text{ in } \mathcal{A}(f) \\ &= p \in b \text{ in } (\mathcal{A}(f))^T. \end{aligned}$$

□

Note that this duality can be used to prove that f^* is bent, using the standard design-theoretic Proposition 9.3 to do the hard work. Given f we define $\mathcal{A}(f)$, use Proposition 9.4 or 9.5 to show that it is a square 2-design, use Proposition 9.3 to show that its dual is also a square 2-design, and then prove converses to Propositions 9.4 and 9.5.

We have shown that $\mathcal{A}(f)$ is a 2-design, but in fact it is almost a 3-design. For a design to be a 3-design, it must be the case that the intersection of any 3 blocks has constant size. Given that it is a 2-design this is equivalent to demanding that the symmetric difference of any 3 blocks has constant size. Thus the next best thing is that these various symmetric differences have only two sizes. In fact we can demand rather more than this, as follows:

Definition: A square 2-design $\mathcal{D}$ is said to have the *symmetric difference property* or be an *SDP design* if, given any three distinct blocks B_1, B_2, B_3 of the design, their symmetric difference $B_1 \triangle B_2 \triangle B_3$ is either a block of the design or the complement of a block. □

Clearly since all the blocks of the design have the same size this condition implies that the symmetric differences of 3 blocks do indeed have only two sizes.

Proposition 9.10: *If f is a bent function then $\mathcal{A}(f)$ is an SDP design.*

Proof: Consider three distinct blocks $f + {b_1}^?$, $f + {b_2}^?$, $f + {b_3}^?$ of $\mathcal{A}(f)$. Their symmetric difference is

$$f + {b_1}^? + f + {b_2}^? + f + {b_3}^? = f + {b_1}^? + {b_2}^? + {b_3}^? = f + (b_1 + b_2 + b_3)^?$$

which is either the block indexed by $b_1 + b_2 + b_3$ or its complement. □

SDP designs are considered by Parker, Spence and Tonchev [33], particularly the case $n = 6$. They note the above result and also its converse, quoting Dillon and Schatz:

Theorem 9.11: *If $\mathcal{D}$ is an SDP design then it is the addition design of a bent function.*

Proof: Let n be minimal such that $V(n, 2)$ has at least as many points as $\mathcal{D}$ has blocks and let $\{e_1, \ldots, e_n\}$ be a basis for V. Label the blocks of $\mathcal{D}$ with V-points as follows: pick arbitrary distinct blocks B_0, B_{e_1}, B_{e_2}. Then $B_0 \triangle B_{e_1} \triangle B_{e_2}$ is a block up to complementation — label this block $B_{e_1+e_2}$.

Now proceed inductively — suppose we have labelled some set of blocks with labels from a subspace $R = \langle e_1, \ldots, e_r \rangle$ in such a way that

$$B_0 \triangle B_x \triangle B_y \doteqdot B_{x+y} \quad \forall x, y \in R. \qquad (*)$$

Then either $r = n$ or we can pick a new block and label it $B_{e_{r+1}}$. Now for each $x \in R$ we know that $B_0 \triangle B_x \triangle B_{e_r}$ is a block up to complementation — label this block $B_{x+e_{r+1}}$. Property $(*)$ ensures that these blocks are distinct from each other and from those which already have labels, since no two blocks can be complementary. It can also be shown that we still have $(*)$ even if x and y now range throughout $\langle R, e_{r+1} \rangle$ — this can be shown by an induction on the weight of the lighter of x and y, for example.

When we have completed this labelling we will have labelled the blocks of $\mathcal{D}$ using all the points of V (since we label $2^?$ blocks at each stage), so $\mathcal{D}$ has 2^n blocks and hence 2^n points. Now we label the points of $\mathcal{D}$ with V-points – we start by picking a point and labelling it 0. For each point P of the design, label it with the V-point

$$\sum_{1 \leqslant i \leqslant n} ([P \in B_0 \triangle B_{e_i}] \ \hat{+} \ [0 \in B_0 \triangle B_{e_i}])\, e_i.$$

This formula means that $B_0 \triangle B_{e_i}$ is just the support of ${e_i}^?$. As before an induction on the weight of a label shows that more generally if $x \neq 0$ then $B_0 \triangle B_x$ is the support of $x^?$.

This means that $2k - 2\lambda = |B_0 \triangle B_x| = 2^{n-1}$ so using Proposition 9.1 and the fact that the design is square we can show that the design's parameters must be those of an addition design of a bent function on V.

Furthermore we know that if $x \neq 0$ then $B_0 \triangle B_x \doteq x^{\bar{\perp}} \implies B_0 \triangle x^{\bar{\perp}} \doteq B_x$, so $|B_0 \triangle x^{\bar{\perp}}| = 2^{n-1} \pm 2^{m-1}$, so by Proposition 1.10 B_0 is (the support of) a bent function on V and $\mathcal{D}$ is its addition design. □

This result is also proved by Kantor [17]. Note that Theorem 9.11, saying that a design with appropriate properties must be induced by a bent function, is similar to Theorem 2.4, which did the same thing for codes with appropriate properties. However Theorem 2.4 was easier, since there we assumed that the code was linear and so got the relationship with V for free, whereas in Theorem 9.11 we have proved it from scratch.

Note also that the addition design $\mathcal{A}(f)$ cannot actually be a 3-design (unless $n = 2$). To see this we attempt to calculate what the parameter λ would be by counting pairs consisting of a block and a set of 3 points contained in it in two ways. We find

$$b\binom{k}{3} = \binom{v}{3}\lambda \implies \lambda = 2^{n-3} \pm 2^{m-1} \mp \frac{2^{3m-4}}{2^{n-1}-1}$$

which implies that λ is an integer only if $n = 2$. Thus for larger n both possible weights of the symmetric difference of 3 blocks must actually occur.

10. Automorphisms of designs

A design automorphism is just what one would expect:

Definition: An *isomorphism* between designs $\mathcal{D} = (\mathcal{P}, \mathcal{B}) \to \mathcal{E} = (\mathcal{Q}, \mathcal{C})$ is a pair of bijections $\pi : \mathcal{P} \to \mathcal{Q}$ and $\rho : \mathcal{B} \to \mathcal{C}$ which respect the incidence relation, i.e. such that for all $P \in \mathcal{P}, B \in \mathcal{B}$, P is incident with B in $\mathcal{D}$ iff πP is incident with ρB in $\mathcal{E}$.

An *automorphism* of a design $\mathcal{D}$ is an isomorphism from $\mathcal{D}$ to itself. □

So let A, B be the adjacency matrices of $\mathcal{D}$, $\mathcal{E}$ — if π acts on the (indices of the) columns of A and ρ acts similarly on its rows then this action takes A to B.

If the designs are simple then the point action π uniquely specifies the block action ρ, so we often talk of "the isomorphism π" when we mean "the isomorphism (π, ρ)". The set of isomorphisms π from a design $\mathcal{D}$ to itself forms a group under composition, the automorphism group of $\mathcal{D}$, which we write $\mathrm{Aut}(\mathcal{D})$.

We now need to consider the automorphism group of an SDP design $\mathcal{A}(f)$. Although in general the points $\mathcal{P}$ of a design are just a set, in this case they are the points of our underlying vector space V, so we start by considering linear automorphisms of this space. We saw in Chapter 3 that, given a basis for V, we can represent a linear automorphism as a matrix. Thus we write α^T to mean the automorphism corresponding to the transpose of α's matrix.

Lemma 10.1: *If* $\alpha \in \mathrm{GL}(V)$ *then*

$$x.y = (\alpha^{-1}x).(\alpha^T y).$$

Proof: Given a basis for V, α has an associated transformation matrix A, which is non-singular. Considering points of V as column vectors of co-ordinates with respect to this basis we have

$$x.y = x^T y = x^T A^{-T} A^T y = (A^{-1}x)^T A^T y = (\alpha^{-1}x).(\alpha^T y).$$

□

We can now prove a close relationship between a linear automorphism applied to f and the associated design isomorphism:

Proposition 10.2: *If* $\alpha \in GL(V)$ *then*

$$(\pi, \rho) = (\alpha, \alpha^{-T}) : \mathcal{A}(f) \to \mathcal{A}(\alpha f)$$

is a design isomorphism.

Proof: Using Theorem 9.6, for all $p, b \in V$ we have

$$\begin{aligned} p \in b \text{ in } \mathcal{A}(f) &= p \in f \ \hat{+}\ b \in f^* \ \hat{+}\ 0 \in f^* \ \hat{+}\ p.b \\ &= \alpha p \in \alpha f \ \hat{+}\ \alpha^{-T} b \in \alpha^{-T} f^* \ \hat{+}\ \alpha^{-T} 0 \in \alpha^{-T} f^* \ \hat{+}\ (\alpha p).(\alpha^{-T} b) \\ &= \pi p \in \alpha f \ \hat{+}\ \rho b \in (\alpha f)^* \ \hat{+}\ 0 \in (\alpha f)^* \ \hat{+}\ (\pi p).(\rho b) \\ &= \pi p \in \rho b \text{ in } \mathcal{A}(\alpha f). \end{aligned}$$

□

We get a similar result by considering the effect of a translation on f and its design:

Proposition 10.3: *If $a \in V$ then*

$$(\pi, \rho) = (\bullet_a, \mathrm{id}) : \mathcal{A}(f) \to \mathcal{A}(f_a)$$

is a design isomorphism.

Proof: As before for all $p, b \in V$ we have

$$\begin{aligned} p \in b \text{ in } \mathcal{A}(f) &= p \in f \ \hat{+}\ b \in f^* \ \hat{+}\ 0 \in f^* \ \hat{+}\ p.b \\ &= p + a \in f_a \ \hat{+}\ b \in (f^* + a^{\bar{\perp}}) \ \hat{+}\ 0 \in (f^* + a^{\bar{\perp}}) \ \hat{+}\ (p + a).(b) \\ &= \pi p \in f_a \ \hat{+}\ b \in (f_a)^* \ \hat{+}\ 0 \in (f_a)^* \ \hat{+}\ (\pi p).b \\ &= \pi p \in b \text{ in } \mathcal{A}(f_a). \end{aligned}$$

□

An *affine* map is a composition of a linear automorphism and a translation, and by Lemma 3.1 any composition of operations of these two types can be written as a single composition. We have just shown that any affine action on f induces a corresponding design isomorphism whose point action is also affine. However, somewhat surprisingly this is all that can happen — the *only* possible point actions are affine maps. To prove this we will need the following characterisation of affine maps:

Proposition 10.4: *If π is a permutation of V then π is an affine transformation iff $\pi(x^{\bar{\perp}})$ is in* RM_1 *for all $x \in V$.*

Proof:

$\Rightarrow$: This is immediate from Lemmas 1.17 and 3.1.

$\Leftarrow$: Given any $x \in V$, by hypothesis $\pi(x^{\bar{\perp}})$ is some $y^?$. Thus we can define a permutation η by $\pi(x^{\bar{\perp}}) = (\eta x)^?$.

Now η is a linear map, since

$$\pi((x+y)^{\bar\perp}) = \pi(x^{\bar\perp} \triangle y^{\bar\perp}) = \pi(x^{\bar\perp}) \triangle \pi(y^{\bar\perp}) = (\eta x)^? + (\eta y)^? = (\eta x + \eta y)^?$$

so that $\eta(x+y) = \eta x + \eta y$. Since η is a permutation this means that it is a linear automorphism of V.

Now as z varies $z \in \pi(x^{\bar\perp}) \ \hat{+} \ z \in (\eta x)^{\bar\perp}$ is a constant k (either 0 or 1). But then $k = (\pi^{-1}z).x \ \hat{+} \ z.(\eta x) = (\pi^{-1}z + \eta^T z).x$. If $x \neq 0$ then this last expression can be constant only if $\pi^{-1}z + \eta^T z$ is constant, so if $w = \eta^T z$ we have

$$\begin{aligned} &\pi^{-1}z + \eta^T z \text{ is constant} \implies z + \pi\eta^T z \text{ is constant} \\ \implies &\eta^{-T}w + \pi w \text{ is constant} \implies \pi w = \eta^{-T}w + \text{constant}, \end{aligned}$$

so π is affine. In fact taking $w = 0$ we see that $\pi(\bullet) = \eta^{-T}(\bullet) + \pi 0$. □

Now using this characterisation we can prove

Theorem 10.5: *If f and g are bent functions and $(\pi, \rho) : \mathcal{A}(f) \to \mathcal{A}(g)$ is a design isomorphism then there exist $\alpha \in \mathrm{GL}(V)$ and $p, b \in V$ such that*

$$\pi(\bullet) = \alpha(\bullet) + p, \qquad \rho(\bullet) = \alpha^{-T}(\bullet) + b \qquad \textit{and} \qquad g = [\alpha, p, b^{(\sigma)}]f$$

where $\sigma = f^*(0) \ \hat{+} \ f^*(\alpha^T b) \ \hat{+} \ b.p \ \hat{+} \ 1$.

Proof: Using Theorem 9.6, for all $q, c \in V$

$$q \in f \ \hat{+} \ f^*(c) \ \hat{+} \ f^*(0) \ \hat{+} \ q.c = \pi q \in g \ \hat{+} \ g^*(\rho c) \ \hat{+} \ g^*(0) \ \hat{+} \ (\pi q).(\rho c)$$

and we can rewrite this equation as

$$\pi q \in \left(\pi f + \pi(c^{\bar\perp}) + g + (\rho c)^{\bar\perp}\right) = f^*(c) \ \hat{+} \ f^*(0) \ \hat{+} \ g^*(\rho c) \ \hat{+} \ g^*(0). \qquad (*)$$

Setting $c = 0$ and writing $b = \rho 0$ this becomes

$$\pi q \in \left(\pi f + g + b^{\bar\perp}\right) = g^*(b) \ \hat{+} \ g^*(0).$$

Now the RHS is independent of q, so we must have

$$\pi f + g = b^? \qquad (**)$$

Substituting back into $(*)$ we get

$$\pi q \in \left(b^? + \pi(c^{\bar\perp}) + (\rho c)^{\bar\perp}\right) = f^*(c) \ \hat{+} \ f^*(0) \ \hat{+} \ g^*(\rho c) \ \hat{+} \ g^*(0).$$

As before the RHS is independent of q so for all $c \in V$ we know that $\pi(c^{\bar{\perp}}) = (\rho c)^{\bar{\perp}} + b^? \in \mathrm{RM}_1$. Thus by Proposition 10.4 π is affine, so setting $p = \pi 0$ we can write $\pi(\bullet) = \alpha(\bullet) + p$ for some $\alpha \in \mathrm{GL}(V)$.

By the dual of this argument ρ is also affine, so we can write $\rho(\bullet) = \beta(\bullet) + b$ for some $\beta \in \mathrm{GL}(V)$.

Now

$$\begin{aligned} & \pi(c^{\bar{\perp}}) = (\rho c)^{\bar{\perp}} + b^? \\ \Longrightarrow\ & (\alpha(c^{\bar{\perp}}))_p = (\beta c + b)^{\bar{\perp}} + b^? = (\beta c)^{\bar{\perp}} + b^{\bar{\perp}} + b^? \\ \Longrightarrow\ & \alpha(c^{\bar{\perp}}) \doteq \beta^{-T}(c^{\bar{\perp}}). \end{aligned}$$

Hence since α and β^{-T} are linear automorphisms we must have $\alpha(c) = \beta^{-T}(c)$. This is true for all c so $\alpha = \beta^{-T}$ and hence $\rho(\bullet) = \alpha^{-T}(\bullet) + b$.

Finally from $(**)$ we have that $g = \pi f + b^{(\sigma)} = (\alpha f)_p + b^{(\sigma)}$ for some $\sigma \in \mathbb{F}_2$. By Corollary 9.8 we must have $|f| = |g|$ and so using Proposition 4.5

$$\begin{aligned} f^*(0) &= \begin{cases} {(\alpha f)_p}^*(b) & \text{if } \sigma = 1 \\ {(\alpha f)_p}^*(b) \ \hat{+}\ 1 & \text{if } \sigma = 0 \end{cases} \\ &= b \in \left(\alpha^{-T} f^* + p^{\bar{\perp}}\right) \ \hat{+}\ \sigma \ \hat{+}\ 1 \\ \Longrightarrow\ \sigma &= f^*(0) \ \hat{+}\ f^*(\alpha^T b) \ \hat{+}\ b.p \ \hat{+}\ 1 \end{aligned}$$

as claimed. □

Corollary 10.6: *If $\mathcal{A}(f) \cong \mathcal{A}(g)$ then f and g are equivalent.* □

Note also that given a point action $\pi(\bullet) = \alpha(\bullet) + p$ we can recover α and p since $p = \pi 0$ and then $\alpha(\bullet) = \pi(\bullet) + p$. Since the design is simple we know that π induces the corresponding block action ρ, so we can recover $b = \rho 0$.

So, by this argument and its dual, given any one of π, ρ and $[\alpha, a, b^{(\sigma)}]$ we can recover the others. Thus we have bijections between three sets: P and B, the point and block actions of design isomorphisms $\mathcal{A}(f) \to \mathcal{A}(g)$, and G, the maps in $\mathrm{GB}(V)$ sending f to g.

If $f = g$ then P, B and G form groups under composition, and in fact G is just the stabiliser in $\mathrm{GB}(V)$ of f. We have some obvious homomorphisms between these groups:

Theorem 10.7: *With the above notation the maps*

$$\begin{array}{ccc} G & \to & P \\ [\alpha, p, b^{(\sigma)}](\bullet) & \mapsto & \alpha(\bullet) + a \end{array} \qquad \textit{and} \qquad \begin{array}{ccc} G & \to & B \\ [\alpha, p, b^{(\sigma)}](\bullet) & \mapsto & \alpha^{-T}(\bullet) + b \end{array}$$

are homomorphisms.

Proof: Immediate from Lemma 3.1 and Theorem 3.4. □

Corollary 10.8: *With the above notation* $P \cong B \cong G = \mathrm{Stab}_{\mathrm{GB}(V)}(f)$. □

Note that Corollary 4.8, Theorem 9.9 and Corollary 10.8 all demonstrate the close link between the following correspondences:

$$\begin{array}{ccc} f & \to & f^* \\ \mathcal{A} & \to & \mathcal{A}^T \\ [\alpha, p, b^{(\sigma)}] & \to & [\alpha, p, b^{(\sigma)}]^*. \end{array}$$

Finding the addition designs' automorphism groups

Corollary 10.8 says that finding the automorphism group of $\mathcal{A}(f)$ is equivalent to finding $G = \mathrm{Stab}_{\mathrm{GB}(V)}(f)$. To do this we can use the stabiliser graph and various of the other structures discussed in Chapter 8.

First we wish to find those αs in $\mathrm{GL}(V)$ which could possibly appear in an element $[\alpha, ?, ?^?]$ of G. By Theorem 3.4 these form a group, which we write $A(f)$ or just A. We proceed in stages, starting with the whole of $\mathrm{GL}(V)$ and weeding out elements by a series of tests. The author used GAP [25] for most of these calculations.

Test 1 In Chapter 8 we saw that applying a linear automorphism α to V induces a graph isomorphism from $\Gamma(f)$, the stabiliser graph of f, to $\Gamma(\alpha f)$ and the other equivalences leave $\Gamma(f)$ unchanged. Thus maps in A must certainly be automorphisms of $\Gamma(f)$. Similarly the inverse transposes of these elements must be automorphisms of $\Gamma(f^*)$. In practice it is simpler to select those elements which fix setwise the $D_i(f)$s (the vertices of $\Gamma(f)$ of each degree — see Chapter 8), then select those whose transposes fix the $D_i(f^*)$s, and then go back to select the graph automorphisms.

Test 2 If $[\alpha, p, b^{(\sigma)}] \in G$ then $f = (\alpha f)_p + b^{(\sigma)} \implies (f + b^{(\sigma)})_p = \alpha f$. Now f has degree at least 2, so $f + b^{(\sigma)}$ has the same highest degree terms as f. In Chapter 6 we observed that translating a function of degree at least 2 also preserves the terms of highest degree. Thus $(f + b^{(\sigma)})_p$ also has the same highest degree terms as f. Thus for α to be a candidate for A, if we apply α to these highest degree terms we must get a function with the same highest degree terms.

Again we can dualise this result — α^{-T} must preserve the highest degree terms of f^*.

Test 3 We can use other structures from Chapter 8, for example the k-ary relation $\mathcal{R}$ array or the k-dimensional array A, in a similar way to Test 1.

Note that we do not need to test every α at every stage. We know that the set of αs passing each test forms a subgroup of $\mathrm{GL}(V)$, so if we have some αs which

pass a given test and which generate H we need test only a transversal (set of coset representatives) for H in $\mathrm{GL}(V)$. Either we will find some new αs which pass the test or we will have shown that H contains all the αs which pass, so contains all of A.

For most $n = 8$ cases Tests 1 and 2 are enough — we obtain a group H which we know contains A and this is usually small enough to allow us to test for $\alpha \in A$ directly. We can do this by testing whether $(\alpha f)_p + f \in \mathrm{RM}_1$ for each possible choice of p (see below).

For example, consider class 8.54. $|GL(V)| = 5,348,063,769,211,699,200$. Of these maps $2,097,152$ fix the $D_i(f)$s setwise and are such that their inverse transposes fix the $D_i(f^*)$s setwise. $131,072$ of these are automorphisms of $\Gamma(f)$, and all of these (as it happens) are the inverse transposes of automorphisms of $\Gamma(f^*)$, so pass the whole of Test 1. $4,096$ of these pass Test 2, the high-degree terms test. We can then test these to show $|A| = 8$. Some other $|A|$s are shown in Appendix C.

Some functions do require Test 3, but most of these tests are too expensive if our intermediate group H is large. Even with Tests 1 and 2 we can arrive at an H which is too large to find any element passing the test by random search. Sometimes we can guess an element (for example we can sometimes find elements fixing the high-degree terms of f by inspection and then test whether they are in our H) and hence start the transversal process. However, even then our transversal may be too large and we admit defeat. This is the reason for the inequalities in Appendix C.

Our last test on the elements of A consisted of finding a p such that $(\alpha f)_p + f \in \mathrm{RM}_1$. If b is the point indexing this RM_1 function then we know that $[\alpha, p, b^{(\sigma)}]$ is in G. Note that if f is bent then for given values of α and p there can be only one choice of $b^{(\sigma)}$ such that $[\alpha, p, b^{(\sigma)}] \in G$ — indeed, we know $b^{(\sigma)} = (\alpha f)_p + f$.

However, we do not have to find all possible p for each α in order to obtain the whole of G:

Proposition 10.9: *With the above notation let T be the (possibly trivial) subgroup of translations in P. Identifying T with the T-orbit of 0 we have*

(i) $T = D_{n-1}(f) \cup \{0\}$ *(so this use of T coincides with that of Chapter 8).*

(ii) *The maps in P associated with a given $\alpha \in A$ are precisely $\{\alpha(\bullet) + a + t : t \in T\}$ for some fixed $a \in V$.*

(iii) *Each orbit of P is a union of cosets of T.*

(iv) $|P| = |A|(d_{n-1}(f) + 1)$.

(v) $T \lhd P$.

Proof: Note that T consists of translations through the vectors $\{t : [\mathbb{I}_n, t, ?^?] \in G\}$.

(i) If $t \in T$ then by Theorem 10.5 $f = f_t + ?^?$ so $f_{\langle t \rangle} \in \mathrm{RM}_1$, and conversely, hence we have the result by Lemma 6.8.

(ii) If $\alpha \in A$ then some map $[\alpha, a, b^{(\sigma)}]$ is in G so $f = (\alpha f)_a + b^{(\sigma)}$. As in (i) if $t \in T$ then $f = f_t + c^{(\tau)}$ for some c, τ. Equating these RHSs and using Lemma 3.1 we see that $f = (\alpha f)_{a+t} + ?^?$ and hence that $\alpha(\bullet) + a + t \in P$.

Conversely if two maps $\alpha(\bullet) + a_1$ and $\alpha(\bullet) + a_2$ are in P then for $i = 1, 2$ we have $f = (\alpha f)_{a_i} + ?^? \implies f_{a_i} = \alpha f + ?^?$. Thus $f_{a_1} = f_{a_2} + ?^? \implies f = f_{a_1+a_2} + ?^?$ and hence $a_1 + a_2 \in T$.

(iii) and (iv) are immediate from (ii).

(v) is a straightforward check using Theorems 3.4 and 10.7. □

Note that we can also apply the dual of this result to show that each block orbit is a union of cosets of T^*, and $T^* \triangleleft B$.

Corollary 10.10: *The P-orbit of the 0th point and the B-orbit of the 0th block are the same size.*

Proof: These orbits "are" just T and T^*, so we have a bijection between them by Theorem 8.8. □

Thus we can find T easily using Proposition 10.9(i), and if we know the point actions $\alpha(\bullet) + p$ corresponding to a set of αs generating A we have only to add a set of generators for T and we have a generating set for P. We can find generators for B (and hence for G) similarly. We can then find the orbits of P and B. Some details of these calculations are given in Appendix C.

Lander [24 Chapter 3] shows that for a general square design a pair (π, β) of point and block actions have the same cycle structure, and the groups P and B have the same number of orbits. Moreover P and B are isomorphic groups, and the isomorphism takes a point action π to the corresponding block action β — we proved this for SDP designs in Corollary 10.8. However for a general square design the *actions* of P and B need not be isomorphic — this is the case for our 2-$(7, 3, 1)$ design, for example, since the stabiliser of a point is transitive on the remaining 6 points but has orbits of size 3 and 4 on the blocks.

In the case of SDP designs we have a canonical labelling of the points and blocks, so if the actions of P and B are isomorphic this means that there is a permutation θ of V which conjugates a point action π to the corresponding block action β. However, this need not be the case. Consider class 8.10, for example, and suppose some permutation θ conjugates P to B as above. Here the point and block actions are transitive, so we can find a set of point actions $\{\pi_v : v \in V\}$ such that $\pi_v(0) = v$, with corresponding block actions $\{\beta_v\}$. Then if $w = \theta(0)$ the points $\{\beta_v(w) : v \in V\}$

must all be distinct. However we can check (by computer) that there is no choice of w for which this is true, so there can be no such permutation θ.

In such cases it is not clear that P and B need have the same orbit sizes. Despite this, in all the examples tested by the author the sizes *are* in fact the same. This is why the tables in Appendix C have only one entry for "Design orbit sizes". Note that by Corollary 10.10 P and B have at least one common orbit size.

Some remarks on the addition designs' automorphism groups

If we have a set of three blocks of $\mathcal{A}(f)$ their symmetric difference has one of two weights, since the design is an SDP design and $k \neq \frac{v}{2}$. On the other hand, both these weights must actually occur, as noted in Chapter 9. Thus the block action cannot be 3-transitive, or even 3-homogeneous, since it cannot take a triple whose symmetric difference has one weight to one whose symmetric difference has the other weight. Since the point action on $\mathcal{A}(f)$ is the block action on $\mathcal{A}(f^*)$ the point action cannot be 3-transitive either.

However the point (or block) action can be 2-transitive — indeed this is always the case for a non-singular quadratic bent function. Since by Corollary 3.9 all the non-singular quadratics are equivalent it is enough to show this for the representative $f = x_1x_2 + \ldots + x_{n-1}x_n$. If we fix some non-zero $v \in V$ we must find a point action in P taking $(0, v)$ to (x, y) for any choice of distinct $x, y \in V$.

Now if α is a linear automorphism such that f and αf have the same terms of degree 2, and a is any vector in V, then $(\alpha f)_a$ differs from f by an RM_1 function, $b^?$ say. Thus $[\alpha, a, b^?] \in \mathrm{Aut}(\mathcal{A}(f))$ so $\alpha(\bullet) + a \in P$. Note that this shows that the group A consists of exactly those maps in $\mathrm{GL}(V)$ preserving the degree-2 terms of f.

So now it is enough to show that the group of αs preserving the degree-2 terms of f is transitive on the non-zero points of V, for then we know we can find α taking v to $x + y$ and so obtain the point action $\alpha(\bullet) + x$ taking $(0, v)$ to (x, y) as required. Thus we have reduced the problem to proving the following:

Lemma 10.11: *The group A of maps in $\mathrm{GL}(V)$ preserving the degree-2 terms of $f = x_1x_2 + \ldots + x_{n-1}x_n$, the symplectic group $\mathrm{Sp}_n(2)$, is transitive on the non-zero points of V.*

Proof:

It is enough to show that we can find a map taking the vector $10\ldots0$ to any other vector z. We will consider maps in terms of their actions on co-ordinates in the pairs $\{1, 2\}, \{3, 4\}, \ldots, \{n-1, n\}$. Starting from $10\ldots0$ we construct our target vector z in four stages, using various maps in A at each stage.

Stage 1 Use maps of the form

$$x_i \mapsto x_i + x_1,$$
$$x_2 \mapsto x_2 + x_{i+1}$$

(where $\{i, i+1\}$ is one of the above pairs) to make the correct number of pairs contain a non-zero entry.

Stage 2 Use maps of the form $x_{i+1} \mapsto x_{i+1} + x_i$ (where $\{i, i+1\}$ is one of the above pairs) to make the correct number of pairs contain 2 non-zero entries.

Stage 3 Use permutations of the pairs to put the pairs of each weight in the correct places.

Stage 4 Use transpositions of the entries within each pair to get the pairs containing a 1 and a 0 the right way round. □

Corollary 10.12: *If f is a non-singular quadratic function then* $\mathrm{Aut}(\mathcal{A}(f))$ *is 2-transitive on points and blocks.* □

Parker, Spence and Tonchev [33] consider bent functions with $n = 6$ and study the automorphism groups of the associated SDP designs in some detail. They find, for example, that each of the four groups has a transitive regular subgroup, and provide explicit generators. This suggests that perhaps the point action is always (1-)transitive. However, many of the $n = 8$ classes provide counterexamples, for example class 8.5, which has two point orbits of sizes 64 and 192. An extreme case of this is class 8.70, which has *trivial* automorphism group (this has been checked explicitly by the author using a C program). Many other classes, for example 8.48, 8.64 and 8.95, have very small groups.

11. More about designs

A second design induced by a bent function

If f is a bent function on V then as well as the addition design $\mathcal{A}(f)$ considered in Chapter 10 there is a second 2-design $\mathcal{T}(f)$ associated with a bent function, whose blocks are just the translates of f:

Proposition 11.1: *If f is light then $\mathcal{T}(f)$ is a square 2-design with parameters*

$$2\text{-}(2^n, 2^{n-1} - 2^{m-1}, 2^{n-2} - 2^{m-1}).$$

Proof: Clearly v and k are as claimed, since all translates f_a of f have the same weight as f.

If $a, b \in V$ then $|f_a \cap f_b| = |(f \cap f_{a+b})_a| = |f \cap f_{a+b}| = 2^{n-2} - 2^{m-1}$ by Lemma 1.12 and Theorem 1.19, so we have a 2-design.

The 2^n translates f_a are distinct, for $f_a = f_b \implies f = f_{a+b} \implies a + b \in \mathrm{Stab}(f) = \{0\}$ by Proposition 6.9. Thus the design is square. □

We call this design the *translation design* of f (hence the notation $\mathcal{T}(f)$). We can prove a heavy version in exactly the same way:

Proposition 11.2: *If f is heavy then $\mathcal{T}(f)$ is a square 2-design with parameters*

$$2\text{-}(2^n, 2^{n-1} + 2^{m-1}, 2^{n-2} + 2^{m-1}).$$

□

As with the addition design we can index the blocks with the points of V in an obvious way, and obtain an explicit incidence relation:

Theorem 11.3: *If f is a bent function and $p, b \in V$ then in $\mathcal{T}(f)$*

$$p \in b = p + b \in f.$$

Proof: $p \in b = p \in f_b = p + b \in f$. □

Corollary 11.4: $\mathcal{T}(f)^T = \mathcal{T}(f)$. □

Corollary 11.5: *The 0th block of $\mathcal{T}(f)$ is f.* □

We also have results about the action of linear automorphisms and translations similar to those obtained for addition designs:

Theorem 11.6: *If $\alpha \in \mathrm{GL}(V)$ then*

$$(\pi, \rho) = (\alpha, \alpha) : \mathcal{T}(f) \to \mathcal{T}(\alpha f)$$

is a design isomorphism.

Proof: Using Theorem 11.3, for all $p, b \in V$ we have

$$\begin{aligned} p \in b \text{ in } \mathcal{T}(f) &= p + b \in f = \alpha p + \alpha b \in \alpha f \\ &= \pi p \in \rho b \text{ in } \mathcal{T}(\alpha f). \end{aligned}$$

□

Theorem 11.7: *If $a \in V$ then*

$$(\pi, \rho) = (\bullet_a, \mathrm{id}) : \mathcal{T}(f) \to \mathcal{T}(f_a)$$

is a design isomorphism.

Proof: As before for all $p, b \in V$ we have

$$\begin{aligned} p \in b \text{ in } \mathcal{T}(f) &= p + b \in f = p + a + b \in f_a \\ &= \pi p \in b \text{ in } \mathcal{T}(f_a). \end{aligned}$$

□

On the other hand there seems to be no easy converse result as there was in the case of addition designs.

Although they have the same parameters, addition designs and translation designs seem somewhat different. For example the automorphism group of a translation design is clearly transitive on its points, since for any $a \in V$ the map

$$(\pi, \rho) = (\bullet_a, \bullet_a) : \mathcal{T}(f) \to \mathcal{T}(f)$$

is a design automorphism, whereas we saw in Chapter 10 that this certainly need not be true of an addition design. On the other hand both types of design vary considerably for various f, so it is somewhat surprising that in fact the two types can *never* coincide if we regard the point and block labels as significant:

Theorem 11.8: *If f, g are bent functions then we cannot have $\mathcal{A}(f) = \mathcal{T}(g)$.*

Proof: Suppose this equality is possible. Then since, by Corollaries 9.8 and 11.5, the 0th block of each design is the associated bent function we must have $f = g$. But then for all pairs of points $p, b \in V$ we have

$$\begin{aligned} & f(p) \ \hat{+} \ f^*(b) \ \hat{+} \ f^*(0) \ \hat{+} \ p.b = p + b \in f \\ \Longrightarrow \ & f(p) \ \hat{+} \ f^*(b) \ \hat{+} \ f^*(0) \ \hat{+} \ p.b \ \hat{+} \ f(p+b) = 0. \end{aligned}$$

So in particular this is true for the pair $p + b, b$ also, so

$$f(p+b) \ \hat{+} \ f^*(b) \ \hat{+} \ f^*(0) \ \hat{+} \ (p+b).b \ \hat{+} \ f(p) = 0.$$

But if we equate the LHSs everything cancels out except $b.b = 0$, and this cannot be true for all $b \in V$. □

This result means that $\mathcal{A}(f)$ and $\mathcal{T}(g)$ cannot even be equivalent via an affine transformation of V acting on the points and blocks, since this would be equivalent to applying this transformation to g to produce another bent function h, and we would then have $\mathcal{A}(f) = \mathcal{T}(h)$.

On the other hand it is certainly possible for designs of the different types, even those induced by the same bent function f, to be isomorphic via general permutations, i.e. by relabelling the points or blocks. Indeed for a quadratic bent function this *always* happens:

Theorem 11.9: *If a bent function f has degree 2 then $\mathcal{A}(f) \cong \mathcal{T}(f)$.*

Proof: Proposition 6.4 says that $f_{\langle v \rangle}$ has lower degree than f, so if f has degree 2 then $f_{\langle v \rangle}$ is some function $l^? \in \mathrm{RM}_1$ and so $f_v = f + l^?$. Thus any block f_v of $\mathcal{T}(f)$ is also a block of $\mathcal{A}(f)$, so the two designs coincide. □

Note that by Corollary 10.12 this means that $\mathrm{Aut}(\mathcal{T}(f))$ is 2-transitive. In fact in this case $\mathcal{A}(f) \cong \mathcal{T}(f)$ is the symplectic square 2-design $S^\epsilon(2m)$, described by Lander [24 Chapter 3], for example. Here $\epsilon = \pm 1$ according as f is heavy or light. Kantor [17, 18] proves a number of results about this design, including various characterisations. For example:

Proposition 11.10: *If $\mathcal{D}$ is a square 2-design then $\mathrm{Aut}(\mathcal{D})$ is 2-transitive on blocks and contains a non-trivial element fixing at least $\frac{v}{2}$ points iff $\mathcal{D}$ is $S^\epsilon(2m)$ for some m, ϵ.* □

Proposition 11.11: *If $\mathcal{D}$ is a square 2-design and some subgroup of $\mathrm{Aut}(\mathcal{D})$ has a regular normal subgroup and is such that the stabiliser of any block B is transitive on B and $\overline{B}$ then $\mathcal{D}$ is $S^\epsilon(2m)$ for some m, ϵ.* □

Derived and residual designs

If $\mathcal{D} = (\mathcal{P}, \mathcal{B})$ is a design and $P \in \mathcal{P}$ then we can define two designs with point and block sets which are subsets of those of $\mathcal{D}$ and incidence induced by that of $\mathcal{D}$, as follows (see [7] or [24], for example):

Definition:

$\mathcal{D}_P$, the *point-derived design with respect to P*, has point set $\mathcal{P} \setminus \{P\}$ and block set $\{B \setminus \{P\} : B \in \mathcal{B}, P \in B\}$.

$\mathcal{D}^P$, the *point-residual design with respect to P*, has point set $\mathcal{P} \setminus \{P\}$ and block set $\{B : B \in \mathcal{B}, P \notin B\}$. □

Thus $\mathcal{D}_P$ is just the blocks of $\mathcal{D}$ which contain P, with P deleted, and $\mathcal{D}^P$ is the blocks of $\mathcal{D}$ not containing P.

It is clear that if $\mathcal{D}$ is a t-design then $\mathcal{D}_P$ is a $(t-1)$-design, since the blocks containing a set S of $t-1$ points in $\mathcal{D}_P$ correspond to those containing the t points $S \cup \{P\}$ in $\mathcal{D}$, and this number is independent of S. Similarly $\mathcal{D}^P$ is a $(t-1)$-design.

We can also define the duals of these notions, and in fact we will normally refer to these duals as *the* derived and residual designs — given $B \in \mathcal{B}$ we define:

Definition:

$\mathcal{D}_B$, the *(block-)derived design with respect to* B, has point set B and block set $\{B \cap B' : B' \in \mathcal{B}, B' \neq B\}$.

$\mathcal{D}^B$, the *(block-)residual design with respect to* B, has point set $\overline{B}$ and block set $\{\overline{B} \cap B' : B' \in \mathcal{B}, B' \neq B\}$. □

Note that the code $\mathcal{C}(f)$ induced by a bent function f is just the union of $\mathcal{A}(f)_f$ and its complement, in the sense that the words of the code are the incidence vectors of the blocks — recall from Corollary 9.8 that f is a block of $\mathcal{A}(f)$ so this makes sense. To see this, recall that a word w of $\mathcal{C}$ corresponds to an RM_1 function $l^{(\sigma)}$, and has 1s in positions corresponding to those points in f which are also in $l^{(\sigma)}$. Thus w has 1s at points in $f \cap l^{(\sigma)} = f \cap (f + l^{(\sigma \hat{+} 1)})$, which is some block of the union of $\mathcal{A}(f)_f$ and its complement.

In general we cannot say much about these designs $\mathcal{D}_B$ and $\mathcal{D}^B$. However, by the symmetry of the definitions $\mathcal{D}_B = ((\mathcal{D}^T)_P)^T$ for corresponding B and P, and similarly for the residual design. Thus if $\mathcal{D}$ is a *square* 2-design then so is its dual and hence $\mathcal{D}_B$ and $\mathcal{D}^B$ are the duals of 1-designs. In fact in this case we can say much more than this — see Lander [24 Chapter 1]:

Theorem 11.12: *If $\mathcal{D}$ is a 2-(v, k, λ) design then*

(i) *$\mathcal{D}_B$ is a 2-$(k, \lambda, \lambda - 1)$ design.*

(ii) *$\mathcal{D}^B$ is a 2-$(v-k, k-\lambda, \lambda)$ design.*

Proof:

(i) By hypothesis $|B \cap C|$ in $\mathcal{D}$ is independent of the choice of C, so $\mathcal{D}_B$ is a 1-design.

Now pick any 2 distinct points of $\mathcal{D}_B$. Since $\mathcal{D}$ is a 2-design they are in λ common blocks of $\mathcal{D}$. However, they must be in B, so they are in $B \cap C$ for $\lambda - 1$ choices of $C \neq B$, i.e. they are in $\lambda - 1$ common blocks of $\mathcal{D}_B$. Thus $\mathcal{D}_B$ is a 2-design. The number of points is $|B| = k$ and each block contains $|B \cap C| = \lambda$ points.

(ii) is proved in essentially the same way — $|\overline{B} \cap C|$ is independent of the choice of C so $\mathcal{D}^B$ is a 1-design, and then 2 distinct points of $\mathcal{D}^B$ are in $\overline{B} \cap C$ for all λ

choices of block C containing them. The number of points is $|\overline{B}| = v - k$ and each block contains $|\overline{B} \cap C| = |C| - |B \cap C| = k - \lambda$ points. □

In Chapter 9 we saw how, given a light bent function f of $n = 2m$ variables, we could form its addition design $\mathcal{A}(f)$, a square 2-design with parameters

$$2\text{-}(2^n, 2^{n-1} - 2^{m-1}, 2^{n-2} - 2^{m-1}).$$

So given a block B we can form $\mathcal{A}_B(f)$ with parameters

$$2\text{-}(2^{n-1} - 2^{m-1}, 2^{n-2} - 2^{m-1}, 2^{n-2} - 2^{m-1} - 1)$$

and $\mathcal{A}^B(f)$ with parameters

$$2\text{-}(2^{n-1} + 2^{m-1}, 2^{n-2}, 2^{n-2} - 2^{m-1}).$$

For example in the $n = 8$ case $\mathcal{A}(f)$ has parameters 2-$(256, 120, 56)$, $\mathcal{A}_B(f)$ has parameters 2-$(120, 56, 55)$, and $\mathcal{A}^B(f)$ has parameters 2-$(136, 64, 56)$.

We also saw that if a 2-design has the SDP then it is almost a 3-design, because the symmetric differences of 3 blocks have only 2 possible weights. There is a similar notion involving symmetric differences of 2 blocks:

Definition: A 2-design is *quasisymmetric* if there are only two possible weights x and y for the intersection of two distinct blocks. □

Recall that if a 2-design is square then its transpose is a 2-design — a quasisymmetric 2-design is one whose transpose is almost a 2-design. Although "square" and "symmetric" both seem reasonably popular, for some reason "quasisymmetric" seems to be used rather than "quasisquare". The point of introducing this notion is:

Proposition 11.13: *If a bent function f has addition design $\mathcal{A}(f)$ and B is a block of this design then the derived design $\mathcal{A}_B(f)$ and residual design $\mathcal{A}^B(f)$ are quasisymmetric.*

Proof: Jungnickel and Tonchev [16 Lemma 2.1]:

Given X a block of $\mathcal{A}$ with $X \neq B$ write X' for the corresponding block $X \cap B$ of $\mathcal{A}_B$.

If C and D are distinct blocks of $\mathcal{A}$ then by the SDP there is a block E of $\mathcal{A}$ such that $B \triangle C \triangle D \doteq E$. Then

$$E' \doteq (B \triangle C \triangle D) \cap B = (B \cap B) \triangle (C \cap B) \triangle (D \cap B) = B \triangle C' \triangle D' \doteq C' \triangle D'.$$

Thus since $|C' \triangle D'|$ can take only the two values $|E'|$ and $|\overline{E'}|$, by Lemma 1.12 $|C' \cap D'|$ can also take only two values, so $\mathcal{A}_B$ is quasisymmetric.

Replacing B by $\overline{B}$ throughout the above argument proves the result for $\mathcal{A}^B$. □

This result will be useful in the next chapter.

12. A connection with strongly-regular graphs

A regular graph has the property that every vertex has the same degree. A strongly-regular graph has an extra regularity property:

Definition: A graph Γ is a *strongly-regular graph with parameters* (n, k, c, d), or an SRG(n, k, c, d), if it is regular with n vertices each of degree k such that if x, y are distinct vertices of Γ then $x \sim y$ (respectively $x \not\sim y$) $\implies$ there are exactly c (d) vertices adjacent to both x and y. □

It is usual to exclude complete graphs and null graphs. We use parameters c and d instead of the more usual λ and μ to avoid confusion with the 2-design parameter also usually called λ.

For example a pentagon, or C_5, is an SRG$(5, 2, 0, 1)$ — any two adjacent vertices have no common neighbours while any two non-adjacent vertices must be at distance 2 and have exactly 1 common neighbour. Similarly an octahedron is an SRG$(6, 4, 2, 4)$. As a more complicated example consider the lattice graph $L_2(m)$ as follows: its vertex set is $S \times S$ where $S = \{1, \ldots, m\}$ and $(i, j) \sim (k, l)$ iff $i = k$ or $j = l$. Thus the vertices are the points of a square integral lattice and two vertices are adjacent iff they are in the same row and column. $L_2(m)$ is an SRG$(m^2, 2(m-1), m-2, 2)$. See [7] for these and other examples, and also an inclusion-exclusion proof of the following:

Proposition 12.1: *The complement of an* SRG(n, k, c, d) *is an* SRG$(n, n-k-1, n-2k+d-2, n-2k+c)$. □

Shrikhande and Sane [36] point out that we can produce an SRG from a quasisymmetric 2-design. Its vertices are the blocks of the design, and two blocks are adjacent iff they intersect in x points (where x and y are the two possible block intersection weights as before). Shrikhande and Sane [36 Theorem 3.8] find the parameters of this graph, although we only sketch their proof:

Proposition 12.2: *The graph produced from a quasisymmetric* 2-(v, k, λ) *design as above is an* SRG(N, K, c, d) *with*

$$N = b, \qquad K = \frac{k(r-1) - y(b-1)}{x - y}, \qquad c = K + \epsilon_1 + \epsilon_2 + \epsilon_1\epsilon_2, \quad \text{and} \quad d = K + \epsilon_1\epsilon_2,$$

where

$$\epsilon_1 = \frac{r - \lambda - k + y}{x - y} \qquad \text{and} \qquad \epsilon_2 = \frac{y - k}{x - y}.$$

Proof: The quasi-symmetry condition can be written in terms of the adjacency matrices of the design and the graph. It can then be shown that the adjacency matrix of the graph has only 3 distinct eigenvalues $\epsilon_0 = K$, ϵ_1 and ϵ_2, and this is enough to guarantee that it is strongly-regular. The eigenvalues and hence the parameters of the graph can then be calculated. □

Now in our case, with light f, $\mathcal{A}_B(f)$ has $(x, y) = (2^{n-3} - 2^{m-1}, 2^{n-3} - 2^{m-2})$ so its strongly-regular graph $\Gamma_B(f)$ has parameters

$$(2^n - 1, 2^{n-1} - 2, 2^{n-2} - 3, 2^{n-2} - 1)$$

while $\mathcal{A}^B(f)$ has $(x, y) = (2^{n-3} - 2^{m-2}, 2^{n-3})$ so that $\Gamma^B(f)$ has parameters

$$(2^n - 1, 2^{n-1}, 2^{n-2}, 2^{n-2}).$$

As an aside note that $\mathcal{A}^B(f)$'s graph has $c = d$. Thus if we take two copies of its vertex set, call one $\mathcal{P}$ and the other $\mathcal{B}$, and then define an incidence relation on $\mathcal{P} \times \mathcal{B}$ induced by adjacency in the graph, we obtain a 2-design with parameters 2-$(2^n - 1, 2^{n-1}, 2^{n-2})$.

The point of these constructions is that by using the derived and residual designs we have produced two SRGs, and by starting with the complement of f instead we could produce two more (possibly with different parameters). However, these four graphs are in fact the same up to complementation (negation of the incidence relation), as we shall show. As with the addition designs we start by establishing a more convenient formulation of the adjacency relation in the SRG:

Lemma 12.3: *If f is a light bent function and B_a, B_b, B_c are distinct blocks of $\mathcal{A}(f)$ (where $B_a = f + a^?$, etc.) then*

$$|B_a \triangle B_b \triangle B_c| = \begin{cases} 2^{n-1} - 2^{m-1} & \text{if } f^*(a) \hat{+} f^*(b) \hat{+} f^*(c) = f^*(a+b+c) \\ 2^{n-1} + 2^{m-1} & \text{otherwise.} \end{cases}$$

Proof: $\mathcal{A}(f)$ has the SDP, so $B_a \triangle B_b \triangle B_c$ obviously has one of these two weights — we need only check which one. Since f is light, $f^*(0) = 0$ so by Corollary 9.7

$$\begin{aligned} &|B_a \triangle B_b \triangle B_c| \\ =&|f + a^{(f^*(a) \hat{+} f^*(0) \hat{+} 1)} + f + b^{(f^*(b) \hat{+} f^*(0) \hat{+} 1)} + f + c^{(f^*(c) \hat{+} f^*(0) \hat{+} 1)}| \\ =&|f + (a+b+c)^{(f^*(a) \hat{+} f^*(b) \hat{+} f^*(c) \hat{+} 1)}|. \end{aligned}$$

Now $f + (a+b+c)^{(f^*(a+b+c) \hat{+} f^*(0) \hat{+} 1)}$ is a block of the design, hence is light. Thus $B_a \triangle B_b \triangle B_c$ is also light iff

$$f^*(a) \hat{+} f^*(b) \hat{+} f^*(c) \hat{+} 1 = f^*(a+b+c) \hat{+} 1,$$

which gives the result claimed. □

Now we can prove our alternative form of the adjacency relation in the SRG:

Theorem 12.4: *If f is light and $B = B_a = f + a^?$ is a block of $\mathcal{A}(f)$ then the adjacency relation in the SRG Γ_B induced from the derived design $\mathcal{A}_B(f)$ is*

$$r \sim s \qquad \Longleftrightarrow \qquad f^*(a) \;\hat{+}\; f^*(b) \;\hat{+}\; f^*(c) = f^*(a+b+c)$$

where r represents the block $B_a \cap B_b$ with $B_b = f + b^?$ and s corresponds to c similarly.

Proof: This is simply a matter of turning a condition in intersections into one on symmetric differences:

$$\begin{aligned}
r \sim s &\iff |(B \cap B_B) \cap (B \cap B_C)| = x \iff 4|B \cap B_B \cap B_C| = 4x \\
&\iff |B| + |B_B| + |B_C| - |B \triangle B_B| - |B_B \triangle B_C| - |B_C \triangle B| \\
&\qquad + |B \triangle B_B \triangle B_C| = 4x \\
&\iff 3(2^{n-1} - 2^{m-1}) - 3.2^{n-1} + |B \triangle B_B \triangle B_C| = 4(2^{n-3} - 2^{m-1}) \\
&\iff |B \triangle B_B \triangle B_C| = 2^{n-1} - 2^{m-1} \\
&\iff f^*(a) \;\hat{+}\; f^*(b) \;\hat{+}\; f^*(c) = f^*(a+b+c).
\end{aligned}$$

□

Armed with this form of the adjacency relation we can show that our four SRGs are in fact the same up to complementation:

Theorem 12.5:
(i) *$\Gamma^B(f)$ is the complement of $\Gamma_B(f)$.*
(ii) *$\Gamma_B(\overline{f})$ is the complement of $\Gamma_B(f)$.*

Proof: Again these are straightforward checks:
(i) Since $x = 2^{n-3} - 2^{m-2}$ in $\mathcal{A}^B(f)$, in $\Gamma^B(f)$ we have

$$\begin{aligned}
r \sim s &\iff |(\overline{B} \cap C) \cap (\overline{B} \cap D)| = x \iff |\overline{B} \cap C \cap D| = x \\
&\iff |C \cap D| - |B \cap C \cap D| = x \\
&\iff |B \cap C \cap D| = 2^{n-2} - 2^{m-1} - 2^{n-3} + 2^{m-2} \\
&\iff |(B \cap C) \cap (B \cap D)| = 2^{n-3} - 2^{m-2}.
\end{aligned}$$

But this last value is the y of $\mathcal{A}_B(f)$ rather than the x, so this condition is the exact negation of that required for adjacency in $\Gamma_B(f)$.

(ii) The blocks of $\mathcal{A}\left(\overline{f}\right)$ are the complements of the blocks of $\mathcal{A}(f)$, so two blocks of $\mathcal{A}\left(\overline{f}\right)$ have the smaller possible intersection size precisely when the corresponding blocks of $\mathcal{A}(f)$ have the larger possible intersection size, so again the adjacency relation for $\Gamma_B\left(\overline{f}\right)$ is the exact negation of that for $\Gamma_B(f)$. □

In fact it turns out to be more convenient to dualise this entire construction, so that we consider $\mathcal{A}(f^*)$. By Theorem 9.9 the blocks of this design "are" the points of $\mathcal{A}(f)$, i.e. just the points of V. Thus we can pick a point $a \in V$, i.e. a block of $\mathcal{A}(f^*)$, use this block to define a derived design, and hence obtain an SRG on the remaining blocks of $\mathcal{A}(f^*)$, i.e. the points of $V \setminus \{a\}$. For light f by Theorem 12.4 this SRG has adjacency relation

$$r \sim s \qquad \Longleftrightarrow \qquad f(a) \mathbin{\hat{+}} f(r) \mathbin{\hat{+}} f(s) = f(a+r+s).$$

We call this the ath SRG of f. In particular if we take $a = 0$ we obtain an SRG on the non-zero points of V with

$$r \sim s \qquad \Longleftrightarrow \qquad |f \cap \langle r, s\rangle| \text{ is even.}$$

Conversely if we know f's 0th SRG then f is determined as uniquely as we could hope for:

Theorem 12.6: *Two bent functions f and g have the same 0th SRG iff they differ by a function in* RM_1.

Proof:

$\Leftarrow$: If $g = f + l^{(\sigma)}$ with $l^{(\sigma)} \in \mathrm{RM}_1$ and $W = \langle r, s\rangle$ then

$$g \cap W = (f \triangle l^{(\sigma)}) \cap W = (f \cap W) \triangle (l^{(\sigma)} \cap W).$$

Now $l^{(\sigma)}$ has dimension at least $n-1$ and W has dimension 2, so $l^{(\sigma)} \cap W$ is a non-trivial subspace, so contains evenly many points, so $f \cap W$ and $g \cap W$ have the same parity.

$\Rightarrow$: If f and g have the same 0th SRG then let $h = f + g$. Now for any 2-space $W = \langle r, s\rangle$ we have

$$|h \cap W| = |(f+g) \cap W| = |(f \cap W) \triangle (g \cap W)| = |f \cap W| + |g \cap W| - 2|(f \cap g) \cap (g \cap W)|$$

which is even since $|f \cap W|$ and $|g \cap W|$ have the same parity.

So pick any non-zero vector r and let $U = \langle r\rangle$. For any $s \notin U$ the set $U \sqcup U_s$ is a 2-space so $|h \cap (U \sqcup U_s)|$ is even, so $|h \cap U|$ and $|h \cap U_s|$ have the same parity. So translation by r either fixes h (if this parity is even) or complements h (if not). Thus r is either in $\mathrm{Stab}(h)$ or in the coset of it described in Proposition 6.10(ii) if this exists. Since r was arbitrary the union of $\mathrm{Stab}(h)$ and this coset must be the whole of V, so $\mathrm{Stab}(h)$ has dimension at least $n-1$, so h is in RM_1 by Lemma 6.8. □

A class of structures respected by equivalences

Like several structures we have considered previously, these SRGs involve a defining condition which takes a set of points of V and checks whether various linear combinations of them are in f. This is the class of structures mentioned at the end of Chapter 8, which we now investigate.

Consider a vector space $W = V(k, 2)$ with a basis $\{e_1, \ldots, e_k\}$. A point $w \in W$ induces a function $w(\bullet) : V^k \to V$ as follows: write w in terms of the basis vectors as $\sum \lambda_i e_i$ where $\lambda_i \in \mathbb{F}_2$ and then set

$$w(v_1, \ldots, v_k) = \sum \lambda_i v_i$$

(here $\sum$ means addition in V). Thus the co-ordinates of the point w indicate which of the arguments of $w(\bullet)$ are to be added together. So given a subset $S \subseteq W$ and a subset $f \subseteq V$ we can define a function $\varphi(S, f)(\bullet) : V^k \to \mathbb{F}_2$ by

$$\varphi(S, f)(\underline{v}) = \sum_{w \in S} (w(\underline{v}) \in f)$$

(here $\sum$ means addition in $\mathbb{F}_2$).

First we establish some basic properties of the functions w and φ:

Lemma 12.7: *If $w, w' \in W$, $S \subseteq W$ and $f, g \subseteq V$ then*

(i) $(w + w')(\underline{v}) = w(\underline{v}) + w'(\underline{v})$.

(ii) $\varphi(S, f + g) = \varphi(S, f) \ \hat{+} \ \varphi(S, g)$.

(iii) $\varphi\left(S, \overline{f}\right) = \varphi(S, f) \ \hat{+} \ [|S| \text{ is odd}]$.

Proof: These are all fairly straightforward checks:

(i) is true because corresponding cancellations occur in W on the LHS and in V on the RHS.

(ii)
$$\begin{aligned}\varphi(S, f + g)(\underline{v}) &= \sum_{w \in S} (w(\underline{v}) \in (f + g)) = \sum_{w \in S} (w(\underline{v}) \in f \ \hat{+} \ w(\underline{v}) \in g) \\ &= \sum_{w \in S} (w(\underline{v}) \in f) \ \hat{+} \ \sum_{w \in S} (w(\underline{v}) \in g) \\ &= \varphi(S, f)(\underline{v}) \ \hat{+} \ \varphi(S, g)(\underline{v}).\end{aligned}$$

(iii) $\varphi\left(S, \overline{f}\right)(\underline{v}) = \sum_{w \in S} (w(\underline{v}) \notin f) = \sum_{w \in S} (w(\underline{v}) \in f \ \hat{+} \ 1)$ and these extra 1s cancel iff $|S|$ is even. □

Now we can investigate how these functions behave if we apply various generators of GB(V) (acting componentwise on $\underline{v}$):

Lemma 12.8: *With notation as before, if $\alpha \in GL(V)$ and $p \in V$ then*

(i) $\alpha w(\underline{v}) = w(\alpha \underline{v})$.

(ii) *If* $\mathrm{wt}(w)$ *is even then* $w(\underline{v}) = w(\underline{v}_p)$. *If* $\mathrm{wt}(w)$ *is odd then* $w(\underline{v}) + p = w(\underline{v}_p)$.

Proof:

(i) is immediate from the linearity of α.

(ii) In each case on the RHS the translation of the arguments through p adds $\mathrm{wt}(w)$ copies of p to $w(\underline{v})$. □

Corollary 12.9:

(i) $\varphi(S, \alpha f)(\underline{v}) = \varphi(S, f)(\alpha^{-1}\underline{v})$.

(ii) *If* $\mathrm{wt}(w)$ *is odd* $\forall w \in S$ *then* $\varphi(S, f_p)(\underline{v}) = \varphi(S, f)(\underline{v}_p)$. □

We also need the following special case:

Theorem 12.10: *With notation as before, if* $S = T_{\langle x \rangle}$ *with* $x \neq 0$ *and* $l \in V$ *then*

$$\varphi(S, l^{\bar{\perp}})(\underline{v}) = \begin{cases} 1 & \text{if } |T| \text{ is odd and } x(\underline{v}).l = 1 \\ 0 & \text{otherwise.} \end{cases}$$

Proof:

$$\begin{aligned} \varphi(S, l^{\bar{\perp}})(\underline{v}) &= \sum_{w \in S} \left(w(\underline{v}) \in l^{\bar{\perp}} \right) = \sum_{t \in T} \left(t(\underline{v}) \in l^{\bar{\perp}} \ \hat{+} \ (t+x)(\underline{v}) \in l^{\bar{\perp}} \right) \\ &= \sum_{t \in T} \left(t(\underline{v}) \in l^{\bar{\perp}} \ \hat{+} \ t(\underline{v}) \in l^{\bar{\perp}}{}_{x(\underline{v})} \right) = \sum_{t \in T} \left(t(\underline{v}) \in l^{\bar{\perp}}{}_{\langle x(\underline{v}) \rangle} \right) \end{aligned}$$

(the second equality is true because points in $T \cap T_x$ are counted twice so cancel out). But by Lemma 1.17

$$l^{\bar{\perp}}{}_{\langle x(\underline{v}) \rangle} = \begin{cases} \mathbb{1} & \text{if } x(\underline{v}).l = 1 \\ \mathbb{O} & \text{otherwise} \end{cases}$$

so $\varphi(S, l^{\bar{\perp}})(\underline{v})$ is just the parity of $|T|$ in the first case and 0 in the second case. □

So we can now re-prove various earlier results about structures which can be expressed in terms of φ:

Example 1 Recall from Chapter 8 that the stabiliser graph $\Gamma(f)$ of a bent function f is a graph on the points of V with adjacency relation

$$x \sim y \iff f_{\langle x,y \rangle} = \mathbb{O}$$

for x, y distinct and non-zero. We can take $k = 3$, so $W = V \times V \times V$, and $S = \{\cdot\cdot 1, 1\cdot 1, \cdot 11, 111\}$. Then

$$\begin{aligned} \varphi(S, f)(x, y, z) &= z \in f \ \hat{+} \ x + z \in f \ \hat{+} \ y + z \in f \ \hat{+} \ x + y + z \in f \\ &= z \in f \ \hat{+} \ z \in f_x \ \hat{+} \ z \in f_y \ \hat{+} \ z \in f_{x+y} \\ &= z \in f_{\langle x,y \rangle} \end{aligned}$$

so we can rewrite the adjacency relation as $x \sim y \iff \varphi(S,f)(x,y,z) = 0$ for all $z \in V$. Then since $S = \{\cdot\cdot 1, 1\cdot 1\}_{\langle \cdot 1 \cdot \rangle}$, by Lemma 12.7(ii) and Theorem 12.10 we see that adding an RM_1 function to f leaves $\Gamma(f)$ unchanged, as proved in Chapter 8. Similarly the fact that we get an isomorphism from $\Gamma(f)$ to $\Gamma(\alpha f)$ by applying α^{-1} to V follows from Corollary 12.9(i).

Example 2 In the same way we can show that the k-ary relation $\mathcal{R}(v_1, \ldots, v_k)$ defined at the end of Chapter 9 is left invariant by addition of RM_1 functions, and behaves as we expect with respect to linear automorphisms.

We can also use these results to consider applying the point action π of an automorphism of $\mathcal{A}(f)$ componentwise to the argument of φ:

Theorem 12.11: *With notation as before, suppose that $S = T_{\langle x \rangle}$ with $|T|$ even and $\mathrm{wt}(w)$ is odd $\forall w \in S$. Let f be a bent function on V and pick $\pi(\bullet) = \alpha(\bullet) + p \in P(f)$, i.e. π is the point action of an automorphism of $\mathcal{A}(f)$. Then $\varphi(S,f)(\pi \underline{v}) = \varphi(S,f)(\underline{v})$.*

Proof: By Theorem 10.5 $\alpha^{-1}(f_p) = f + b^?$ for some $b^? \in \mathrm{RM}_1$ and hence

$$\begin{aligned}\varphi(S,f)(\pi\underline{v}) &= \varphi(S,f)((\alpha\underline{v})p) = \varphi(S,f_p)(\alpha\underline{v}) = \varphi(S,\alpha^{-1}(f_p))(\underline{v}) \\ &= \varphi(S,f+b^?)(\underline{v}) = \varphi(S,f)(\underline{v}) \;\hat{+}\; \varphi(S,b^?)(\underline{v}) \\ &= \varphi(S,f)(\underline{v}).\end{aligned}$$

□

We have thus shown that any structure that can be suitably defined in terms of φ is respected by the point actions $P(f)$:

Example 3 Consider the SRGs defined in this chapter. Given f we can define a ternary relation

$$\mathcal{R}(a,r,s) = \begin{cases} 1 & \text{if } r \sim s \text{ in the } a\text{th SRG of } f \\ 0 & \text{otherwise.} \end{cases}$$

for r, s distinct and non-zero. Then by the discussion following Theorem 12.5

$$\mathcal{R}(a,r,s) = f(a) \;\hat{+}\; f(r) \;\hat{+}\; f(s) \;\hat{+}\; f(a+r+s) \;\hat{+}\; 1 = \varphi(S,f)(a,r,s) \;\hat{+}\; 1$$

where $S = \{1\cdot\cdot, \cdot 1\cdot, \cdot\cdot 1, 111\} = \{1\cdot\cdot, \cdot 1\cdot\}_{\langle 1\cdot 1\rangle}$. So a point action $\pi \in P(f)$ preserves φ and hence $\mathcal{R}$.

What does this preservation of $\mathcal{R}$ mean? It means that if we apply an automorphism π then $\pi r \sim \pi s$ in the (πa)th SRG of f iff $r \sim s$ in the ath SRG of f. In other words, π is a graph-theoretic isomorphism from the ath SRG to the (πa)th SRG.

Thus the SRGs associated with a bent function f are partitioned into isomorphism classes corresponding to the point orbits of $\mathrm{Aut}(\mathcal{A}(f))$, although graphs corresponding to distinct point orbits may or may not be isomorphic.

13. A connection with partial geometries

As with designs we use the usual definition of a partial geometry — see [7].

Definition: A *partial geometry with parameters* (s, t, α), or $\mathrm{PG}(s, t, \alpha)$, is an incidence structure consisting of pair of sets $(\mathcal{P}, \mathcal{L})$ (the *points* and *lines* of the geometry) and an incidence relation I on $\mathcal{P} \times \mathcal{L}$ such that

(i) Each line in $\mathcal{L}$ is incident with ("contains") exactly $s + 1$ points in $\mathcal{P}$.

(ii) Each point in $\mathcal{P}$ is incident with ("on" or "in") exactly $t + 1$ lines in $\mathcal{L}$.

(iii) Two lines contain at most one point, and two points are contained in at most one line.

(iv) Given any line L and any point P not on L, there are exactly α points on L collinear with P (two points are collinear if there is some line of $\mathcal{L}$ containing both of them). □

Note that, with non-negative parameters, $s + 1$ and $t + 1$ are non-zero so $\mathcal{P}$ and $\mathcal{L}$ must be non-empty.

As a simple example, consider a vertex set $S \times S$ with $|S| = m$, as used to define the lattice graph in Chapter 12. Take the rows and columns as lines. Then this is a $\mathrm{PG}(m - 1, 1, 1)$.

As another example consider the 2-$(7, 3, 1)$ design used to introduce square designs in Chapter 9. Taking the blocks as lines this is a $\mathrm{PG}(2, 2, 3)$.

As with designs and other incidence structures, we have the concept of the *dual* of a partial geometry, obtained by interchanging $\mathcal{P}$ and $\mathcal{L}$. This merely interchanges s and t — note that α is unchanged, since condition (iv) of the definition could equivalently end "... there are exactly α lines through P concurrent with L (two lines are concurrent if there is some point of $\mathcal{P}$ contained in both of them)".

Given a partial geometry we can define a graph on its vertices as follows:

Definition: The *point graph* of a partial geometry has the points of the geometry as its vertices, with two vertices adjacent iff they are collinear in the geometry. □

Note that we lose information in passing from the partial geometry to the graph, since we don't record *which* line contains a given pair of adjacent points. On the other hand we can show that

Theorem 13.1:

(i) *The point graph of a* $\mathrm{PG}(s, t, \alpha)$ *is strongly-regular with parameters*

$$((s + 1)(st + \alpha)/\alpha, (t + 1)s, s - 1 + t(\alpha - 1), (t + 1)\alpha).$$

(ii) *If an* SRG(n, k, c, d) *has parameters of this form then the possible values of* s, t *and* α *are unique.*

Proof:

(i) Cameron and van Lint [7 Theorem 7.3].

(ii) By considering k and d, we see that $t = (k-s)/s$ and $\alpha = ds/k$. So

$$\begin{aligned} & c = s - 1 + \frac{k-s}{s}\left(\frac{ds}{k} - 1\right) \\ \Longrightarrow\ & cks = ks^2 - ks + (k-s)(ds-k) \\ \Longrightarrow\ & 0 = (k-d)s^2 - k(c-d)s - k^2 \\ \Longrightarrow\ & s = \frac{k}{2(k-d)}\left(c - d \pm \sqrt{(c-d)^2 + 4(k-d)}\right). \end{aligned}$$

Now $k > d$ so if we take the negative root here we obtain a negative value for s, which is impossible. Thus there is a unique solution for s and hence for t and α. □

Note that in some cases the point graph may be complete or null, and in this chapter it is convenient to consider these to be SRGs, whereas in Chapter 12 we excluded them.

If an SRG does have parameters of the form described in Theorem 13.1(i) it need not be the point graph of a PG. Even if it is, reconstruction of the PG from the SRG is neither unique, owing to the loss in information in the transition mentioned above, nor easy, since it involves looking for appropriate cliques in the SRG corresponding to lines of the PG.

Given a set of possible parameters for an SRG we can find what those of its complement must be using Proposition 12.1, even if we don't actually know whether a graph with these parameters exists. Similarly given a set of possible PG parameters we can find those of its dual directly by interchanging s and t. Using Theorem 13.1 we can pass between the parameters of a PG and those of its point graph, again without knowing whether either exists.

We can apply these results repeatedly to obtain a chain of possible parameter sets of various objects, although we must stop if a parameter set is obviously impossible (if it contains non-integers or negative numbers, for example).

However, if we start with the ath SRG of a bent function, as defined in Chapter 12, we obtain quite a long sequence of parameter sets which at least look feasible. The sequence ends in each direction because the next PG would have points but no lines, which is impossible. The sequence is as follows — the original SRG is line 6:

1 $\mathrm{SRG}(\quad 2^{n-1}-2^{m-1}, \quad 0, \quad 2^m, \quad 0 \quad)$

↖↘ complement

2 $\mathrm{SRG}(\quad 2^{n-1}-2^{m-1}, \quad 2^{n-1}-2^{m-1}-1,$
$2^{n-1}-2^{m-1}-2, \quad 2^{n-1}+2^{m-1} \quad)$

↖↘ point graph

3 $\mathrm{PG}(\quad 2^{m-1}-1, \quad 2^m, \quad 2^{m-1} \quad)$

↖↘ dual

4 $\mathrm{PG}(\quad 2^m, \quad 2^{m-1}-1, \quad 2^{m-1} \quad)$

↖↘ point graph

5 $\mathrm{SRG}(\quad 2^n-1, \quad 2^{n-1}, \quad 2^{n-2}, \quad 2^{n-2} \quad)$

↖↘ complement

6 $\mathrm{SRG}(\quad 2^n-1, \quad 2^{n-1}-2, \quad 2^{n-2}-3, \quad 2^{n-2}-1 \quad)$

↖↘ point graph

7 $\mathrm{PG}(\quad 2^m-2, \quad 2^{m-1}, \quad 2^{m-1}-1 \quad)$

↖↘ dual

8 $\mathrm{PG}(\quad 2^{m-1}, \quad 2^m-2, \quad 2^{m-1}-1 \quad)$

↖↘ point graph

9 $\mathrm{SRG}(\quad 2^{n-1}+2^m+2^{m-1}+1, \quad 2^{n-1}-2^{m-1},$
$2^{n-1}-2^{m+1}-2^{m-1}+3, \quad 2^{n-1}-2^m-2^{m-1}+1 \quad)$

↖↘ complement

10 $\mathrm{SRG}(\quad 2^{n-1}+2^m+2^{m-1}+1, \quad 2^{m+1}, \quad 2^m, \quad 4 \quad)$

↖↘ point graph

11 $\mathrm{PG}(\quad 2^m, \quad 1, \quad 2 \quad)$

↖↘ dual

12 $\mathrm{PG}(\quad 1, \quad 2^m, \quad 2 \quad)$

↖↘ point graph

13 $\mathrm{SRG}(\quad 2^m+2, \quad 2^m+1, \quad 2^m, \quad 2^{m+1}+2 \quad)$

↖↘ complement

14 $\mathrm{SRG}(\quad 2^m+2, \quad 0, \quad 2^m, \quad 0 \quad)$

Although the number of points of a PG is not recorded as one of its parameters, it is the same as the number of vertices, i.e. the first parameter, of the corresponding SRG. Thus we can easily find the number of points (and, by duality, the number of lines) of the various PGs in this diagram.

We would like to know how much of this diagram actually exists and behaves as expected, for various choices of function f and point a. We define some useful concepts:

Definition: A parameter set (n, k, c, d) is said to be *SRG-feasible* if it satisfies the following conditions:

(i) $k(k-c-1) = (n-k-1)d$.

(ii) $n \geqslant 2k-d+2$ and $n \geqslant 2k-c$.

(iii) The Krein conditions: We can calculate what the eigenvalues of the adjacency matrix of a graph with these parameters would be — they are k and the roots of

$$\tfrac{1}{2}\left(c-d \pm \sqrt{(c-d)^2+4(k-d)}\right).$$

If these roots are p and q then the conditions are that

$$(p{+}1)(k{+}p{+}2pq) \leqslant (k{+}p)(q{+}1)^2 \qquad \text{and} \qquad (q{+}1)(k{+}q{+}2pq) \leqslant (k{+}q)(p{+}1)^2.$$

This condition's derivation requires that the graph and its complement are connected — note that an SRG must be connected if $d > 0$.

(iv) The multiplicities of these eigenvalues, which turn out to be

$$\tfrac{1}{2}\left(n-1 \pm \frac{(n-1)(d-c)-2k}{\sqrt{(c-d)^2+4(k-d)}}\right),$$

must be non-negative integers. □

These conditions are explained in more detail, and proved, by Cameron and van Lint [7 Chapter 2]. (i) is proved by a counting argument, (ii) by considering the complement of the putative graph, and (iii) and (iv) by expressing the strong-regularity conditions in terms of the adjacency matrix as in Proposition 12.2.

All these conditions on the parameters are necessary for them to be the parameters of an SRG. However the choice of conditions is not particularly standardised — one can take various different sets of necessary conditions to obtain different notions of SRG-feasibility.

Using this notion we make the following:

Definition: A parameter set is said to be *pseudo-geometric* if it could feasibly be the parameters of the point graph of a PG, i.e. if it is of the form given in Theorem 13.1 with s, t, α non-negative and integral such that $\alpha \leqslant s+1$, $\alpha \leqslant t+1$.

A parameter set is *strongly pseudo-geometric* if it is pseudo-geometric and the parameter set of the point graph of the dual of its corresponding geometry is SRG-feasible.

A parameter set is *geometric* if there is an SRG with these parameters which is the point graph of some PG.

We say that an SRG with a pseudo-geometric parameter set is pseudo-geometric, and similarly for "strongly pseudo-geometric" and "geometric". □

Now it is a straightforward check that all the SRG parameter sets in the table are SRG-feasible, so they are all strongly pseudo-geometric except 1 and 14.

In fact we can actually pin down most of the entries in the table. First the easy cases:

Proposition 13.2:
(i) 1 *and* 14 *are null graphs.*
(ii) 2 *and* 13 *are complete graphs.*

Proof: Obvious from the parameter sets. □

Definition: For any r, the *triangular graph of order* r, $T(r)$ has as its vertex set the set of unordered pairs of integers from $1, \ldots, r$ with two pairs joined iff they have non-trivial intersection. □

Proposition 13.3: *$T(r)$ is an SRG with parameters*

$$\left(\binom{r}{2}, 2r-4, r-2, 4\right).$$

Proof: Clearly $T(r)$ has $\binom{r}{2}$ vertices. The automorphism group is edge- and non-edge transitive so it is enough to perform the following checks:

$\{1,2\}$ is joined to $r-2$ vertices of the form $\{1,?\}$ with $? \neq 1,2$ and another $r-2$ of the form $\{2,?\}$ with $? \neq 1,2$, so $k = 2r-4$.

$\{1,2\}$ and $\{1,3\}$ are mutually joined to $\{2,3\}$ and $r-3$ vertices of the form $\{1,?\}$ with $? \neq 1,2,3$, so $c = r-2$.

$\{1,2\}$ and $\{3,4\}$ are mutually joined to $\{1,3\}$, $\{2,4\}$, $\{1,4\}$ and $\{2,3\}$, so $d = 4$. □

Hoffman [15] proves that the converse is true for $n \neq 8$, and gives a counterexample with $n = 8$ (an SRG$(28, 12, 6, 4)$ which is not $T(8)$). However as described in [7] we can prove almost all the cases we need using the following result of Bose [4], which is proved by a careful consideration of the cliques (complete subgraphs) and claws (subgraphs with one vertex joined to all the others and no other edges) of the SRG:

Proposition 13.4: *A pseudo-geometric SRG corresponding to* PG(s,t,α) *with*

$$s > \tfrac{1}{2}(t+2)\left((t-1) + \alpha(t^2+1)\right)$$

is geometric. □

We also need the following (see [7 Proposition 7.9], for example):

Lemma 13.5: *If a partial geometry has $\alpha = s + 1$ then its points and lines can be regarded as the points and blocks of a 2-design with $\lambda = 1$.*

Proof: Consider two points P and Q — we must show that there is exactly one line through both of them. There cannot be more than one by condition (iii) of the definition. On the other hand there must be *some* line L through P — if Q is in L then we're done, otherwise there are lines through Q hitting all α of the points of L, one of which must be P. □

Now we can prove the uniqueness of $T(r)$ in almost all cases:

Proposition 13.6: *If $r > 8$ then $T(r)$ is the only SRG with parameters*

$$\left(\binom{r}{2}, 2r-4, r-2, 4\right).$$

Proof: Suppose Γ is a graph with these parameters. By Theorem 13.1 the corresponding PG parameters are $\mathrm{PG}(r-2, 1, 2)$, and if $r > 8$ these parameters satisfy the hypothesis of Proposition 13.4 so Γ is geometric.

So now consider the dual of Γ's PG, with parameters $\mathrm{PG}(1, r-2, 2)$. By Lemma 13.5 we can regard this as a 2-design — by a straightforward calculation its parameters are 2-$(r, 2, 1)$ and it has $\binom{r}{2}$ blocks of size 2. Thus *every* pair of its points must be a block, so it must be the trivial design. This in turn means that Γ must be $T(r)$. □

We also need to prove the uniqueness of $T(r)$ for two smaller values of r:

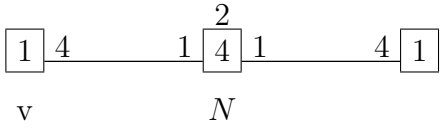
Lemma 13.7:

(i) *$T(4)$ is the only* $\mathrm{SRG}(6, 4, 2, 4)$.

(ii) *$T(6)$ is the only* $\mathrm{SRG}(15, 8, 4, 4)$.

Proof:

(i) Pick a vertex v — this has 4 neighbours, the set N say. Each of these neighbours is joined to $c = 2$ of the others and hence to $4 - 2 - 1 = 1$ vertex at distance 2 from v (in fact there is only 1 vertex left). It is now easy to fill in the rest of the graph's distribution diagram:

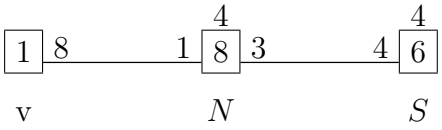


In such a diagram each box represents the vertices at a given distance from v, and the numbers in and around the middle box, for example, mean that there are

4 vertices at distance 1 from v, each joined to 1 of those at distance 0 (although in fact there *is* only one of these), each joined to 2 others at distance 1, and each joined to 1 of those at distance 2 (although in fact again there is only one of these). See Brouwer, Cohen and Neumaier [5] for more details.

Now consider the 4 vertices in the middle set N — the only way each can be joined to 2 others is for them to form a 4-cycle. Hence the graph is $T(4)$. Note that this is the octahedron, which we met in Chapter 12.

(ii) Arguing as before v has 8 neighbours N, each joined to 4 others and hence to 3 vertices at distance 2 from v. On the other hand since $d = 4$ each vertex at distance 2 must be joined to 4 of the 8 vertices at distance 1. Hence counting these edges in two ways there must be 6 vertices at distance 2 — call this set S. This accounts for all 15 vertices, and we can complete the distribution diagram:



It is easy to show that the subgraph induced on S is an SRG$(6, 4, 2, 4)$, so is an octahedron by (i). A pair of antipodes of this octahedron must be joined to disjoint sets of 4 vertices in N, and by considering two such pairs we partition the vertices of N into 4 pairs $P_1, \ldots, P_4$.

Now consider the third pair of antipodes. We can show that either each is joined to every P_i or each is joined to exactly 2 of the P_is. In the first case the subgraph induced on N would be an SRG$(8, 4, 1, 1)$, but this violates condition (ii) of SRG-feasibility, so is impossible. Hence we are in the second case. If we pick a vertex of S we know that the set of vertices not adjacent to it induces an octahedron, and using this we can show that the vertices within each P_i are adjacent and that various pairs of the P_is induce a 4-cycle. It is then easy to confirm that the graph is $T(6)$ as required. □

Corollary 13.8: 10 *is* $T(2^m + 2)$ *and* 12 *can be considered as the trivial* 2-$(2^m + 2, 2, 1)$ *design.* □

Now we know that 10 and hence its complement 9 exist we can ask whether 9 is geometric, i.e. does the PG 8 exist? The general question "Is $\overline{T(r)}$ geometric?" is a difficult one which has been extensively investigated by Thas and others. However the special case $r = 2^m + 2$ is much easier. It involves the notion of a projective plane, which we now explore — this discussion is based on [7 Chapter 1]:

Definition: A *projective plane of order* n is a 2-$(n^2 + n + 1, n + 1, 1)$ design. □

The 2-$(7, 3, 1)$ design, which we have seen several times already, is a projective plane of order 2 (in fact the only one). We shall see other examples later.

Definition: If $\mathcal{D}$ is a square 2-design then an *n-arc* is a set of n points of $\mathcal{D}$ such that no 3 of them are in a common block.

Given an n-arc S a block B of $\mathcal{D}$ is called a *secant*, *tangent* or *passant* to S according as it meets S in 2, 1 or 0 points. □

We have the following useful result:

Proposition 13.9: *Each point of an n-arc in a square 2-(v, k, λ) design lies on $(n-1)\lambda$ secants and $k-(n-1)\lambda$ tangents.*

Proof: Let p be a point in an n-arc S. Count pairs (q, B) where B is a secant to S containing p and q, and $q \in S \setminus \{p\}$, in two ways. □

This allows the following:

Definition: An n-arc S is called an *oval of Type I* if every point of S lies on a unique tangent, and an *oval of Type II* if it has no tangents. □

Corollary 13.10: *An n-arc S is a Type I oval iff $n = 1 + \frac{k-1}{\lambda}$, and it is a Type II oval iff $n = 1 + \frac{k}{\lambda}$.*

Proof: Proposition 13.9. □

Now we need to show that the projective planes we will require actually exist:

Proposition 13.11: *If q is a prime power then there exists a projective plane of order q.*

Proof: To construct the plane we can just generalise the method used to construct the 2-$(7, 3, 1)$ design in Chapter 9:

Consider the vector space $V = V(3, q)$. Define a design $\mathcal{D}$ whose points are the 1-spaces of V and whose blocks are the 2-spaces of V. Say a block is incident with those points it contains.

Now V has $q^3 - 1$ non-zero points and each 1-space contains $q - 1$ of these, so the design has $q^2 + q + 1$ $\mathcal{D}$-points. A $\mathcal{D}$-block, or 2-space, contains $q^2 - 1$ non-zero points of V, so contains $q + 1$ $\mathcal{D}$-points. Finally two distinct 1-spaces are in exactly one 2-space, namely the one they span. □

Note that this projective plane is just the projective geometry of dimension 2 over $\mathbb{F}_q$, if we identify blocks of the plane with lines of the geometry. However the standard notation for this, $\mathrm{PG}(2, q)$, is easily confused with that for a partial geometry, so since we are only interested in this one family of projective planes we

will call this plane Π_q instead. We will show that it has a Type I oval and hence that it has a Type II oval — see [7].

Proposition 13.12: *Π_q has a Type I oval.*

Proof: Consider the set of 0s of the function $x_1x_3 - {x_2}^2$ on the vector space $V(3,q)$ used to define the plane in Proposition 13.11. Since the function is homogeneous this set of 0s must consist of entire 1-spaces of V, and clearly these are the spans of the vectors $(1,t,t^2)$ where $t \in \mathbb{F}_q$, together with $(0,0,1)$. Let S be this set of 1-spaces, considered as Π-points.

Suppose we have 3 distinct Π-points in S which lie in a common Π-block, i.e. lie in a 2-space of V. If one of them is $\langle(0,0,1)\rangle$ then there are non-zero $\lambda, \mu \in \mathbb{F}_2$ such that

$$(0,0,1) = \lambda(1,t_1,{t_1}^2) + \mu(1,t_2,{t_2}^2)$$

and otherwise there are non-zero $\lambda, \mu \in \mathbb{F}_2$ such that

$$(1,t_1,{t_1}^2) = \lambda(1,t_2,{t_2}^2) + \mu(1,t_3,{t_3}^2).$$

In either case we can equate coefficients to obtain 3 simultaneous equations in 2 unknowns which have no solution.

This we have shown that S is a $(q+1)$-arc of Π_q, and by Corollary 13.10 it is a Type I oval. □

So now we use this Type I oval to obtain a Type II oval:

Lemma 13.13: *If S is a Type I oval of a square 2-(v,k,λ) design with $k-\lambda$ even then any point of the design lies on either one or all tangents to S.*

Proof: The existence of a Type I oval implies that $\lambda \mid (k-1)$ so k, λ are both odd. By Corollary 13.10 $n\lambda = \lambda + k - 1$ so if n is even we have even.λ = odd + odd − 1 which is impossible. Hence n is also odd.

Thus every point p not in S lies on at least one tangent to S, since $\lambda \geqslant 1$ so there are lines from p through each point of S and these lines can't all be secants since n is odd.

Now for $1 \leqslant i \leqslant n$ let c_i be the number of points which lie on i tangents to S. Counting various types of pair in two ways we have

$$\sum_i c_i = v, \qquad \sum_i ic_i = nk \qquad \text{and} \qquad \sum_i i(i-1)c_i = n(n-1)\lambda$$

and hence by Proposition 9.1(iv) and Corollary 13.10

$$\sum_i (i-1)(i-n)c_i = 0$$

so that $c_i = 0$ unless $i = 1$ or $i = n$, as claimed. □

Proposition 13.14: *In a projective plane of even order with a Type I oval S, the tangents to S pass through a common point p. Moreover $S \cup \{p\}$ is a Type II oval.*

Proof: In the notation of Lemma 13.13 we have $c_1 + c_n = v$ and $c_1 + nc_n = nk = (q+1)^2 = v+q$, and hence $c_n = 1$, so there is a point p lying on all n tangents to S.

Clearly p cannot be in S, so let $T = S \cup \{p\}$. Suppose 3 points of T lie in a common block B. One of these points must be p since S is already an oval — let the others be x and y. By construction p lies on the tangent C to S through x. Since $\lambda = 1$ B and C must be the same block. But now $y \in C$ so C is a secant to S rather than a tangent.

This we have shown that T is a $(q+2)$-arc of Π_q, and by Corollary 13.10 it is a Type II oval. □

Corollary 13.15: *Π_q has a Type II oval.* □

Now we can produce the partial geometry we need, using a construction due to Wallis [38]:

Proposition 13.16: *If there exists a projective plane Π of order q containing an Type II oval O then $\overline{T(q+2)}$ is geometric.*

Proof: If p is a Π-point not in O then every Π-line through p hits O in either 0 or 2 Π-points (so O has no tangents). Take the PG-points to be unordered pairs of Π-points in O — note that O contains $q+2$ Π-points. For each Π-point p not in O define a PG-line of the geometry containing those PG-points (pairs of Π-points of O) which are on a Π-line through p.

Now the PG-points correspond to the vertices of $\overline{T(q+2)}$. Two PG-points are PG-collinear
iff the Π-lines through them intersect in a Π-point p not in O
iff the Π-lines through them don't intersect in a Π-point of O
iff the PG-points are disjoint (considered as pairs of Π-points)
iff the corresponding vertices of $\overline{T(q+2)}$ are adjacent.

Thus $\overline{T(q+2)}$ is indeed the point graph of this geometry. □

Corollary 13.17: *For all positive even $n = 2m$, $\overline{T(2^m+2)}$ is geometric.* □

Thus we know that $\mathfrak{8}$, and hence $\mathfrak{7}$, exist, not necessarily uniquely. However, we don't know that their point graphs need be the ath SRG of some bent function. Even given the SRG, trying to find a suitable labelling of its vertices with the points of V in an attempt to find the bent function is hard. Similarly although $\mathfrak{8}$ may come

from a projective plane, as described in Proposition 13.16, finding out whether it actually does or not is also a hard problem.

Looking at the other end of the sequence, it is quite easy to show that 3 cannot be the parameter set of a PG:

Theorem 13.18: *There is no* $\mathrm{PG}(2^{m-1}-1, 2^m, 2^{m-1})$.

Proof: Suppose such a geometry exists. Then it has $\alpha = s+1$ so can be regarded as a 2-design with parameters 2-$(2^{n-1}-2^{m-1}, 2^{m-1}, 1)$. This 2-design would have $2^{n-1}-2^{m-1}$ points, 2^n-1 blocks and 2^{m-1} points in each block, so the number of blocks on each point would be

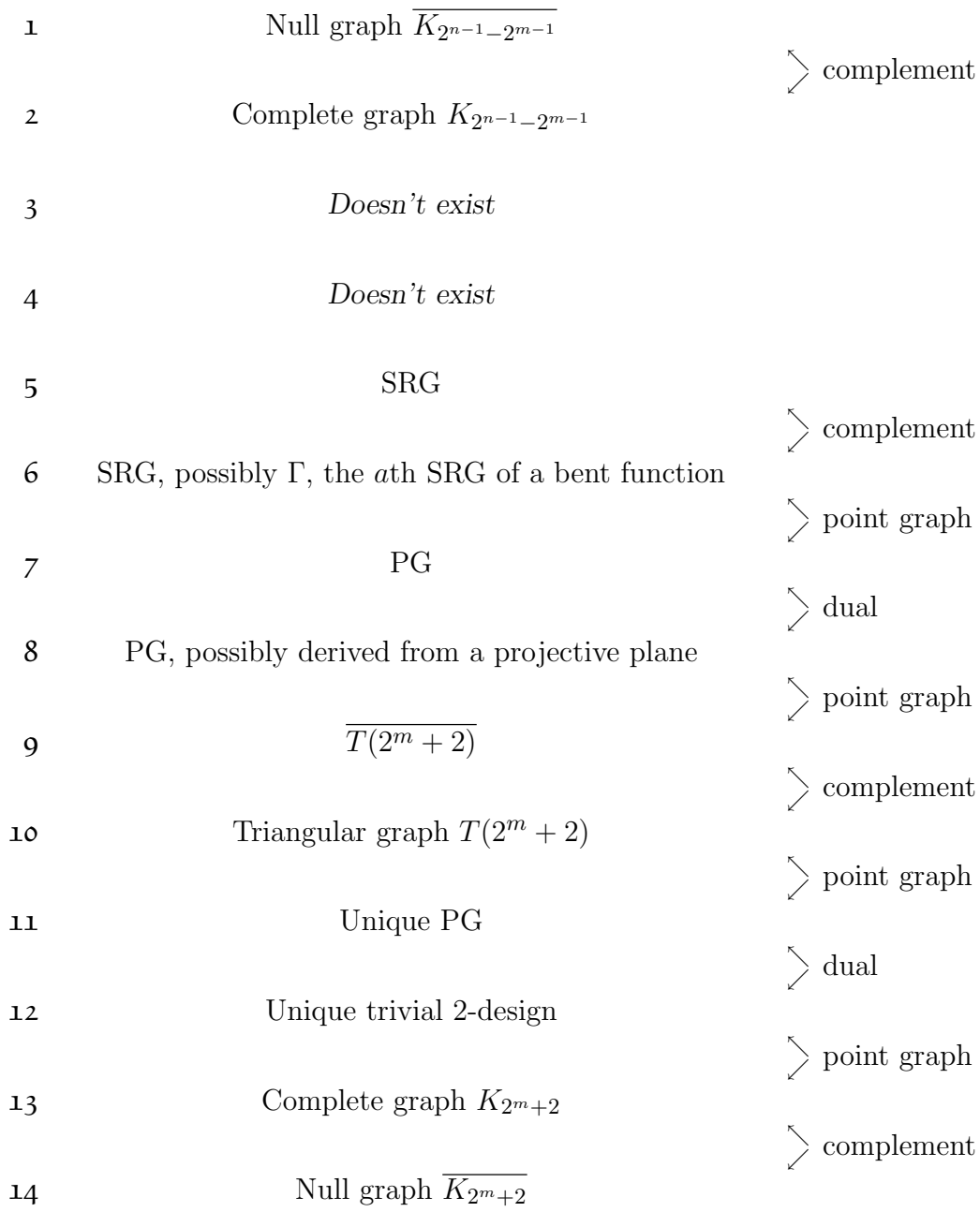
$$\frac{(2^n-1)(2^{m-1})}{2^{n-1}-2^{m-1}} = \frac{2^n-1}{2^m-1} = \frac{(2^{m-1}+1)(2^m-1)-2^{m-1}}{2^m-1} = 2^{m-1}+1-\frac{2^{m-1}}{2^m-1}$$

and this last term cannot be an integer if $m > 0$. □

Various of these results were checked by the author using GRAPE 2.1 [37], Soicher's library of graph functions for GAP [25]. We summarise them in the following proposition:

Proposition 13.19: *If* Γ *is the* a*th SRG of a bent function then* Γ *is strongly pseudo-geometric and* $\overline{\Gamma}$ *is strongly pseudo-geometric but not geometric. If* Γ *is geometric then the dual of its geometry must have* $\overline{T(2^m+2)}$ *as its point graph, so it may come from the projective plane construction.* □

Thus a bent function gives us a strongly-regular graph which is strongly pseudo-geometric but not geometric (or perhaps several such). We can now rewrite the diagram as follows:


1
Null graph $\overline{K_{2^{n-1}-2^{m-1}}}$
complement
2
Complete graph $K_{2^{n-1}-2^{m-1}}$
3
Doesn't exist
4
Doesn't exist
5
SRG
complement
6
SRG, possibly Γ, the ath SRG of a bent function
point graph
7
PG
dual
8
PG, possibly derived from a projective plane
point graph
9
$\overline{T(2^m+2)}$
complement
10
Triangular graph $T(2^m+2)$
point graph
11
Unique PG
dual
12
Unique trivial 2-design
point graph
13
Complete graph K_{2^m+2}
complement
14
Null graph $\overline{K_{2^m+2}}$

14. Self-dual bent functions

In Chapter 8 we saw that a bent function and its dual may or may not be equivalent. However, it is interesting to ask whether there is a bent function equivalent to the original one which is actually *equal* to its dual.

If we apply an equivalence to a bent function f which moves it to a bent function g, Corollary 4.8 tells us the corresponding equivalence which moves f^* to g^*. We would like to know whether we can move f and f^* to the *same* function $s = s^*$ by using a suitable pair of equivalences. Note that given f there may be several self-dual functions s equivalent to it. For example given some self-dual s we can generally permute its variables to produce a second, distinct, self-dual function:

$$x_1x_2 + x_3x_4 \qquad \text{and} \qquad x_1x_3 + x_2x_4$$

are distinct equivalent self-dual bent functions.

Clearly a necessary condition is that f is equivalent to its dual f^*, since each is equivalent to $s = s^*$. In fact if $s = [\beta, c, d^{(\tau)}]f$ then using Corollary 4.8

$$s^* = s \iff [\beta, c, d^{(\tau)}]^* f^* = [\beta, c, d^{(\tau)}]f \iff f^* = [\beta, c, d^{(\tau)}]^{-*}[\beta, c, d^{(\tau)}]f.$$

Note that $[\beta, c, d^{(\tau)}]^{-*}$ is unambiguously $[\beta^T, \beta^T d, (\beta^{-1}c)^{(\tau)}]$ — this is a straightforward check using Theorem 3.4.

If f and f^* are equivalent then the set of maps taking f to f^* is a coset of those stabilising f. Thus we want to find a map $[\alpha, a, b^{(\sigma)}]$ in this coset which we can write in the form $[\alpha, a, b^{(\sigma)}] = [\beta, c, d^{(\tau)}]^{-*}[\beta, c, d^{(\tau)}]$ — we can then find our self-dual function $s = [\beta, c, d^{(\tau)}]f$.

Lemma 14.1: $[\beta, c, d^{(\tau)}]^{-*}[\beta, c, d^{(\tau)}] = [\beta^T\beta, \beta^T(c+d), \big(\beta^{-1}(c+d)\big)^{(d.d \,\hat{+}\, 1)}]$.

Proof: This is a straightforward check:

$$\begin{aligned}
[\beta, c, d^{(\tau)}]^{-*}[\beta, c, d^{(\tau)}] &= [\beta^T, \beta^T d, (\beta^{-1}c)^{(\tau)}][\beta, c, d^{(\tau)}] \\
&= [\beta^T\beta, \beta^T d + \beta^T c, (\beta^{-1}c + \beta^{-1}d)^{(\tau \,\hat{+}\, \tau \,\hat{+}\, \beta^T d.\beta^{-1}d \,\hat{+}\, 1)}] \\
&= [\beta^T\beta, \beta^T(c+d), \big(\beta^{-1}(c+d)\big)^{(d.d \,\hat{+}\, 1)}].
\end{aligned}$$

□

So if $[\alpha, a, b^{(\sigma)}]$ is to be of this form we must have $\alpha = \beta^T\beta$. This tells us quite a lot about α:

Proposition 14.2: *A matrix α over $\mathbb{F}_2$ can be written in the form $\alpha = \beta^T\beta$ with β non-singular iff α is non-singular, symmetric and has a 1 somewhere on its main diagonal.*

Proof: This is all immediate from Proposition 3.6 since $\mathbb{I}_n$ has the stated properties. □

In addition to $\alpha = \beta^T\beta$, by Lemma 14.1 we also require that

$$a = \beta^T(c+d) \qquad \text{and} \qquad b = \beta^{-1}(c+d).$$

Thus a necessary condition is that $a = \beta^T\beta b = \alpha b$. Since we can then pick c and d such that $c+d = \beta^{-T}a$ and $d.d \;\hat{+}\; 1 = \sigma$ our necessary conditions are also sufficient.

Thus we want on our map $[\alpha, a, b^{(\sigma)}]$ to have the following properties, which we call the *self-duality properties*:

(i) α is non-singular, symmetric and has non-zero diagonal.

(ii) $a = \alpha b$.

To summarise and check the above discussion we have

Lemma 14.3: *If $\alpha = \beta^T\beta$, $b = \alpha^{-1}a$, $c = \beta^{-T}a + d$ and $d.d \;\hat{+}\; 1 = \sigma$ then*

$$[\beta, c, d^{(\tau)}]^* f^* = [\beta, c, d^{(\tau)}]f \qquad \Longleftrightarrow \qquad f^* = [\alpha, a, b^{(\sigma)}]f.$$

Proof: This is a straightforward check if we note that $b = \beta^{-1}(c+d)$:

$$\begin{aligned}
&[\beta, c, d^{(\tau)}]^* f^* = [\beta, c, d^{(\tau)}]f \\
&\quad\Longleftrightarrow\; f^* = [\beta, c, d^{(\tau)}]^{-*}[\beta, c, d^{(\tau)}]f \\
&\quad\Longleftrightarrow\; f^* = [\beta^T\beta, \beta^T(c+d), \left(\beta^{-1}(c+d)\right)^{(d.d \;\hat{+}\; 1)}]f \\
&\quad\Longleftrightarrow\; f^* = [\alpha, a, b^{(\sigma)}]f.
\end{aligned}$$

□

Theorem 14.4: *f is equivalent to a self-dual function iff $f^* = [\alpha, a, b^{(\sigma)}]f$ for some $[\alpha, a, b^{(\sigma)}]$ with the self-duality properties.*

Proof:

$\Rightarrow$: If f is equivalent to some self-dual function s then $s = [\beta, c, d^{(\tau)}]f$ for some β, c, d, τ with β non-singular. Let $\alpha = \beta^T\beta$, $a = \beta^T(c+d)$, $b = \alpha^{-1}a$ and $\sigma = d.d \;\hat{+}\; 1$. Then by Lemma 14.3 we have $f^* = [\alpha, a, b^{(\sigma)}]f$. This map satisfies the self-duality properties by Proposition 14.2 and the definition of b.

$\Leftarrow$: By Proposition 14.2 we can write $\alpha = \beta^T\beta$. Pick $d \in V$ such that $d.d \;\hat{+}\; 1 = \sigma$ and then let $c = \beta^{-T}a + d$. Let $s = [\beta, c, d^{\bar{\perp}}]f$. Then by Lemma 14.3 $s = s^*$ so f is equivalent to the self-dual function s. □

We say that such a map $[\alpha, a, b^{(\sigma)}]$ is a *Theorem 14.4 map for f*.

Note that τ is not actually used much in Theorem 14.4 — this is because if f is equivalent to a self-dual function s it is also equivalent to the self-dual function $\overline{s}$.

Clearly if f is equivalent to some self-dual function s then the same is true of any other function g in f's equivalence class. Now if $g = [\gamma, p, q^{(\upsilon)}]f$, and as before $f^* = [\alpha, a, b^{(\sigma)}]f$, then

$$g^* = [\gamma, p, q^{(\upsilon)}]^* f^* = [\gamma, p, q^{(\upsilon)}]^* [\alpha, a, b^{(\sigma)}]f = [\gamma, p, q^{(\upsilon)}][\alpha, a, b^{(\sigma)}][\gamma, p, q^{(\upsilon)}]^{-1} g.$$

On the other hand we have

Proposition 14.5: *If $[\alpha, a, b^{(\sigma)}], [\gamma, p, q^{(\upsilon)}] \in \mathrm{GB}(V)$ and $[\alpha, a, b^{(\sigma)}]$ has the self-duality properties then so does $[\gamma, p, q^{(\upsilon)}]^* [\alpha, a, b^{(\sigma)}][\gamma, p, q^{(\upsilon)}]^{-1}$.*

Proof: First we evaluate the product, although we don't need to keep track of the "exponents":

$$\begin{aligned}
&[\gamma, p, q^?]^* [\alpha, a, b^?][\gamma, p, q^?]^{-1} \\
=&[\gamma^{-T}, q, p^?][\alpha, a, b^?][\gamma^{-1}, \gamma^{-1}p, (\gamma^T q)^?] \\
=&[\gamma^{-T}, q, p^?][\alpha\gamma^{-1}, a + \alpha\gamma^{-1}p, (b + \alpha^{-T}\gamma^T q)^?] \\
=&[\gamma^{-T}\alpha\gamma^{-1}, q + \gamma^{-T}a + \gamma^{-T}\alpha\gamma^{-1}p, (p + \gamma b + \gamma\alpha^{-T}\gamma^T q)^?].
\end{aligned}$$

Now α is non-singular, symmetric and has non-zero diagonal, hence $\gamma^{-T}\alpha\gamma^{-1}$ also has these properties. Since α is symmetric, $\alpha\alpha^{-T} = \alpha\alpha^{-1} = \mathbb{I}$, and by hypothesis $a = \alpha b$, so

$$\begin{aligned}
\gamma^{-T}\alpha\gamma^{-1}\left(c + \gamma b + \gamma\alpha^{-T}\gamma^T d\right) &= \gamma^{-T}\alpha\gamma^{-1}c + \gamma^{-T}\alpha b + d \\
&= \gamma^{-T}\alpha\gamma^{-1}c + \gamma^{-T}a + d
\end{aligned}$$

and we have the second self-duality property also. □

Corollary 14.6: *If $g = [\gamma, p, q^{(\upsilon)}]f$ and $[\alpha, a, b^{(\sigma)}]$ is a Theorem 14.4 map for f then $[\gamma, p, q^{(\upsilon)}]^* [\alpha, a, b^{(\sigma)}][\gamma, p, q^{(\upsilon)}]^{-1}$ is a Theorem 14.4 map for g.* □

Corollary 14.7: *If $[\alpha, a, b^{(\sigma)}]$ is a Theorem 14.4 map for f then the Theorem 14.4 maps for f are*

$$\left\{ [\delta, k, l^{(\varphi)}]^* [\alpha, a, b^{(\sigma)}][\delta, k, l^{(\varphi)}]^{-1} : [\delta, k, l^{(\varphi)}] \in \mathrm{Stab}_{\mathrm{GB}}(f) \right\}.$$

□

Corollary 14.8: *If $g = [\gamma, p, q^{(\upsilon)}]f$ and $[\alpha, a, b^{(\sigma)}]$ is a Theorem 14.4 map for f then the Theorem 14.4 maps for g are*

$$\left\{ [\gamma, p, q^{(\upsilon)}]^* [\delta, k, l^{(\varphi)}]^* [\alpha, a, b^{(\sigma)}][\delta, k, l^{(\varphi)}]^{-1} [\gamma, p, q^{(\upsilon)}]^{-1} : [\delta, k, l^{(\varphi)}] \in \mathrm{Stab}_{\mathrm{GB}}(f) \right\}.$$

□

Corollary 14.9: *Any two equivalent functions have the same number of Theorem 14.4 maps.* □

This result is somewhat surprising, for suppose f is not self-dual but some g equivalent to f is self-dual. Then for every γ in $A(g)$ we know that γ^T is in $A(g)$ also (see Chapter 10). Hence we might expect that symmetric matrices would be more common in $A(g)$ than in $A(f)$, so g might have more maps satisfying the self-duality properties.

It turns out that for $n \leqslant 6$ every bent function is equivalent to a self-dual function. For example the following are self-dual class representatives:

2.1: x_1x_2

4.1: $x_1x_2 + x_3x_4$

6.1: $x_1x_2 + x_3x_4 + x_5x_6$

6.2: $x_1x_2x_4 + x_1x_2x_6 + x_1x_4x_5 + x_1x_5x_6 + x_2x_3x_4 + x_2x_3x_6 + x_3x_4x_5 + x_3x_5x_6$
$+x_1x_4 + x_1x_5 + x_2x_3 + x_2x_4 + x_3x_4 + x_4x_5 + x_4x_6 + x_3 + x_5 + x_6$

6.3: $x_1x_2x_4 + x_1x_3x_6 + x_2x_4x_5 + x_3x_5x_6 + x_1x_5 + x_1x_6 + x_2x_3 + x_3x_6 + x_4x_6$
$+x_5x_6 + x_1 + x_2 + x_3 + x_5 + x_6$

6.4: $x_1x_2x_5 + x_1x_3x_4 + x_1x_3x_5 + x_1x_4x_5 + x_1x_4x_6 + x_1x_5x_6 + x_2x_3x_4 + x_2x_3x_5$
$+x_2x_3x_6 + x_2x_4x_6 + x_2x_5x_6 + x_3x_4x_6 + x_1x_2 + x_1x_4 + x_1x_5 + x_1x_6$
$+x_2x_4 + x_2x_6 + x_3x_4 + x_5x_6 + x_1.$

For larger n some classes have self-dual representatives, for example the representative of class 8.1 in Appendix A. On the other hand we saw in Chapter 8 that many bent functions are not equivalent to their duals, so they certainly cannot be equivalent to a self-dual function.

15. Open problems and conclusions

Typical bent functions

It seems likely that the methods used to find the bent functions in Appendix A produce only a small proportion of possible bent functions, as noted in Chapter 7. Moreover the ones which they do produce tend to have nice properties, for example their stabiliser graphs Γ are often non-null, and they have relatively large automorphism groups.

Thus a major problem is to find other, more general, bent functions. Even a single "random" bent function could well provide counter-examples to several of the speculations below. One could work to understand the direct-summand construction more thoroughly, or try to generalise it, but probably any explicit construction such as this may only ever be able to find a small proportion of bent functions. On the other hand a random search does not seem very feasible.

An example of this problem of known bent functions being atypical occurred in the discussion following Corollary 10.10 concerning the point actions P and block actions B of $\mathrm{Aut}(\mathcal{A}(f))$. We saw that although the actions of P and B need not be isomorphic, in all the cases tested their orbits did in fact have the same sizes. This is rather unexpected, and may well not hold for a typical bent function. The bijection between the 0th orbits (Corollary 10.10) would seem to be useful in investigating this, except that we suspect that typically $T = T^* = \{0\}$ (see Chapter 8).

As another example, consider the question of whether *every* finite group can appear as the automorphism group of an SDP design. Kantor [19] shows that every finite group is the automorphism group of some square 2-design, and there are a number of similar results for other combinatorial structures such as graphs.

Now initially the groups of known SDP designs, found in Chapter 10, seem to have properties which might allow us to restrict the possible groups. For example their orders tend to contain a power of 2. However, this is partly because $T \lhd P$ (Proposition 10.9(v)), which is no help if T is trivial. Similarly other nice properties seem to due to the fact that our bent function are atypical, and it may well be that every finite group can indeed occur. Note that given a group we can sometimes use our knowledge of the structure of G, P and B (Theorem 10.5) to place a lower bound on the dimension n of any possible associated SDP design, since we know things about the structure of $GL(V)$.

Kerdock codes

This discussion of Kerdock codes is based on that of Cameron and van Lint [7 Chapter 12].

From Chapter 2 we know that the minimum distance between any two words of the first-order Reed-Muller RM_1 on $V(n,2)$ is 2^{n-1}. Clearly the same is true of any coset of RM_1 considered as a non-linear code. Suppose we construct a code by taking the union of appropriate cosets of RM_1, since then at least we know that the minimum distance between words of each coset is quite large. For simplicity we restrict to cosets in RM_2.

Each coset of RM_1 in RM_2 corresponds to a unique homogeneous quadratic form Q on V, obtained by deleting the linear and constant terms of any coset representative. Suppose two cosets correspond to Q_1 and Q_2, and let $Q = Q_1 - Q_2$. Then the distance between two words, one in each coset, is just the weight of some function in Q's coset. By Corollary 6.6 this is at least $2^{n-1} - 2^{n-r-1}$, where Q has rank $2r$.

Thus to ensure that the overall minimum distance is as large as possible we want Q to have as large a rank as possible. Thus ideally we want the pairwise differences $Q = Q_1 - Q_2$ to be non-singular. Each Q_i corresponds to a symmetric matrix with zero diagonal as in Chapter 3, so for non-singularity all these matrices must have distinct first rows, so there are at most 2^{n-1} of them. Hence we make the following:

Definition: A *Kerdock set* is a set of 2^{n-1} bilinear forms on $V(n,2)$ such that the difference of any two forms in the set is non-singular.

A *Kerdock code* consists of the cosets of RM_1 corresponding to the forms in a Kerdock set. □

Thus a Kerdock code has $2^{n-1}2^{n+1} = 2^{2n}$ words and minimum distance $2^{n-1} - 2^{m-1}$. In general it is non-linear. Kerdock sets are known to exist — Cameron and van Lint [7] give details of a construction due to Dillon, Dye and Kantor which involves spreads on the quadric associated with a quadratic form.

However, to get the minimum distance of this code correct, the only important point about Q was that the functions $Q + l^?$ in its coset had weight at least $2^{n-1} - 2^{m-1}$. Thus instead we could consider a Kerdock-like set of functions on V with the property that the difference of any two functions is a general bent function rather than necessarily a non-singular quadratic. This would then lead to a Kerdock-like code. However, our new code might have more words than a Kerdock code or might be linear with the same parameters, either of which would be an interesting improvement.

Generalised bent functions

Generalising from the $\mathbb{F}_2$ case, quite a lot of work has been done on bent functions on free modules over the ring $\mathbb{Z}_q$ for general q — see Kumar, Scholtz and Welch [23] or Nyberg [31], for example. Such bent functions allow the construction of bent sequences with symbols from $GF(p)$ for general prime p, generalising ideas of Olsen, Scholtz and Welch [32].

The basic definition generalises fairly easily — a function is bent if its Fourier coefficients have unit magnitude. The equivalences of Proposition 3.2 go through, as does the construction which appeared as Method 4 in Chapter 6.

Moreover the Maiorana construction (Proposition 4.2) goes through unchanged, giving generalised bent functions for all q and all even n. Kumar, Scholtz and Welch [23] give constructions for $q \not\equiv 2 \pmod 4$ and $n = 1$, and then use Method 4 to give generalised bent functions for these q and for all n.

On the other hand they show that no generalised bent function exists if $q \equiv 2 \pmod 4$, $q \neq 2$, n is odd and there exists b such that $2^b \equiv -1 \pmod{\frac{q}{2}}$. Note that the $q = 2$, odd n case is ruled out by Proposition 1.7. For this last case Meier and Staffelbach [30] suggest a different generalisation, namely that the Fourier coefficients should have two distinct magnitudes (whereas in Chapter 1 we insisted that they could have only one magnitude). They show that the distance of such a function from the set of affine functions is nearly as good as that for a bent function in the even n case (see Chapter 2), although surprisingly it is not necessarily the best possible. They also give a construction, similar to the four-function construction, involving bent functions on two complementary halfspaces.

Since various special cases of the direct-summand construction seem to generalise so successfully, it would be interesting to know whether the construction itself also generalises. Even being able to generalise Method 5 could provide many new generalised bent functions.

Relationships between structures

In earlier chapters we studied a number of relationships between various functions, codes, SRGs and geometries, and Calderbank and Kantor [6] note some other interesting connections. Several of these seem worth exploring further.

For example, in Chapter 11 we noted that the code induced by a bent function in Chapter 2 is the union of the derived design $\mathcal{A}(f)_f$ and its complement, and there is an obvious correspondence between words of the code and blocks of the design. Can we say anything about the automorphism groups of either structure as a result?

Similarly in Chapter 12 we saw that a bent function f can be reconstructed

from its 0th SRG up to addition of RM_1 functions, provided that we are given the labelling of the vertices of the SRG by the points of V. Can we still do this without being given the labelling? We may have choices to make in assigning the labels — if so, are all the bent function obtained equivalent? Can we write down conditions on an SRG which allow us to say whether it is the 0th SRG of some bent function, and if so can we use this to find new bent functions?

Appendix A. Classes of bent functions with $n = 8$

This list gives a representative of each class and an indication of how it may be constructed — see Chapter 6 for details. Note that often several different methods can be used to construct functions in a given equivalence class. Only the variables' subscripts are shown, so for example 12 represents the monomial x_1x_2. For representatives found by Method 6 (e.g. class 8.37) the numbers in parentheses are (n_r, n_s). In several cases the function listed is not the actual one found by the Method, but a simpler representative of the same class.

8.1: Non-singular quadratic
$f = 12 + 34 + 56 + 78$

8.2: Method 2
$f = 135 + 12 + 34 + 56 + 78$

8.3: Method 5
$f = 1358 + 135 + 12 + 34 + 56 + 78$

8.4: Method 5
$f = 136 + 157 + 12 + 13 + 34 + 56 + 57 + 78$

8.5: Method 5
$f = 1358 + 1368 + 135 + 12 + 34 + 56 + 78$

8.6: Method 5
$f = 123 + 156 + 246 + 13 + 14 + 23 + 24 + 25 + 36 + 78$

8.7: Method 5
$f = 1246 + 1247 + 1346 + 1347 + 127 + 137 + 156 + 157 + 168 + 178 + 246 + 346 + 12 + 34 + 56 + 78$

8.8: Method 5
$f = 1357 + 1567 + 135 + 347 + 367 + 457 + 567 + 12 + 34 + 56 + 78$

8.9: Method 5
$f = 1358 + 1468 + 135 + 12 + 34 + 56 + 78$

8.10: Method 5
$f = 1246 + 1247 + 1346 + 1347 + 127 + 156 + 157 + 168 + 178 + 246 + 346 + 12 + 13 + 34 + 56 + 78$

8.11: Method 5
$f = 2468 + 123 + 138 + 156 + 246 + 248 + 258 + 268 + 358 + 368 + 13 + 14 + 23 + 24 + 25 + 36 + 78$

8.12: Method 5

$f = 1238 + 1568 + 2468 + 128 + 138 + 148 + 238 + 248 + 258 + 348 + 368 + 568 + 12 + 34 + 56 + 78$

8.13: Method 5

$f = 1238 + 1268 + 1568 + 2468 + 126 + 138 + 238 + 248 + 14 + 25 + 36 + 78$

8.14: Method 5

$f = 1238 + 1358 + 1568 + 2468 + 123 + 128 + 138 + 148 + 156 + 238 + 246 + 248 + 258 + 348 + 368 + 568 + 13 + 14 + 23 + 24 + 25 + 36 + 78$

8.15: Method 5

$f = 1356 + 1456 + 1568 + 3456 + 4568 + 123 + 126 + 128 + 134 + 135 + 136 + 145 + 148 + 236 + 268 + 345 + 358 + 368 + 458 + 14 + 17 + 18 + 35 + 37 + 38 + 45 + 78 + 7$

8.16: Method 5

$f = 1356 + 1456 + 1568 + 3456 + 4568 + 123 + 124 + 128 + 135 + 145 + 168 + 234 + 248 + 345 + 358 + 368 + 458 + 16 + 17 + 18 + 35 + 37 + 38 + 45 + 78 + 7$

8.17: Dual of 8.16

$f = 1237 + 1278 + 2347 + 2378 + 2478 + 123 + 127 + 128 + 167 + 238 + 357 + 367 + 457 + 578 + 16 + 24 + 27 + 28 + 35 + 58 + 67 + 68 + 78 + 8$

8.18: Method 5

$f = 1356 + 1456 + 1568 + 3456 + 4568 + 123 + 125 + 128 + 134 + 145 + 168 + 235 + 258 + 345 + 348 + 368 + 458 + 16 + 17 + 18 + 34 + 37 + 38 + 45 + 78 + 7$

8.19: Method 5

$f = 1358 + 4568 + 128 + 168 + 258 + 456 + 568 + 16 + 25 + 34 + 78$

8.20: Method 5

$f = 1358 + 4568 + 128 + 158 + 268 + 456 + 568 + 15 + 26 + 34 + 78$

8.21: Method 5

$f = 1358 + 4568 + 128 + 158 + 248 + 348 + 368 + 456 + 568 + 15 + 24 + 36 + 78$

8.22: Method 5

$f = 1358 + 4568 + 128 + 168 + 248 + 348 + 358 + 456 + 568 + 16 + 24 + 35 + 78$

8.23: Method 5

$f = 1358 + 4568 + 128 + 148 + 258 + 348 + 368 + 456 + 568 + 14 + 25 + 36 + 78$

8.24: Method 5

$f = 1358 + 2468 + 246 + 12 + 34 + 56 + 78$

8.25: Method 5

$f = 1358 + 2468 + 246 + 348 + 368 + 458 + 568 + 12 + 36 + 45 + 78$

8.26: Method 5
$f = 1358 + 2468 + 128 + 168 + 238 + 246 + 348 + 458 + 568 + 16 + 23 + 45 + 78$

8.27: Method 5
$f = 1246 + 1346 + 2468 + 3468 + 123 + 124 + 125 + 126 + 128 + 134 + 135 + 136 + 138 + 156 + 246 + 248 + 258 + 268 + 348 + 358 + 368 + 12 + 14 + 17 + 23 + 24 + 25 + 36 + 78$

8.28: Method 5
$f = 1246 + 1345 + 2468 + 3458 + 123 + 124 + 128 + 134 + 138 + 156 + 246 + 248 + 348 + 12 + 14 + 17 + 23 + 24 + 25 + 36 + 78$

8.29: Method 5
$f = 1234 + 1238 + 1246 + 1268 + 1456 + 1568 + 2348 + 2468 + 4568 + 124 + 126 + 128 + 134 + 138 + 246 + 248 + 348 + 12 + 14 + 17 + 18 + 23 + 24 + 25 + 36 + 78 + 8$

8.30: Method 5
$f = 1234 + 1238 + 1246 + 1258 + 1456 + 1568 + 2348 + 2468 + 4568 + 124 + 126 + 128 + 134 + 135 + 136 + 138 + 246 + 248 + 258 + 268 + 348 + 358 + 368 + 12 + 14 + 17 + 18 + 23 + 24 + 25 + 36 + 78 + 8$

8.31: Method 5
$f = 1234 + 1238 + 1456 + 1568 + 2348 + 4568 + 14 + 17 + 18 + 23 + 26 + 35 + 78 + 8$

8.32: Method 5
$f = 1238 + 1246 + 1345 + 1568 + 2468 + 3458 + 135 + 156 + 358 + 568 + 14 + 17 + 18 + 23 + 26 + 35 + 78 + 8$

8.33: Method 5
$f = 1238 + 1268 + 1458 + 1568 + 123 + 124 + 126 + 135 + 138 + 145 + 148 + 156 + 238 + 248 + 358 + 568 + 13 + 17 + 18 + 23 + 26 + 35 + 78 + 8$

8.34: Dual of 8.33
$f = 2347 + 2367 + 3457 + 4567 + 137 + 147 + 234 + 247 + 267 + 357 + 378 + 456 + 457 + 478 + 14 + 26 + 35 + 37 + 48 + 56 + 78 + 7$

8.35: Method 5
$f = 1468 + 1568 + 2468 + 123 + 138 + 148 + 156 + 158 + 246 + 258 + 13 + 14 + 23 + 24 + 25 + 36 + 78$

8.36: Method 5
$f = 2468 + 123 + 138 + 156 + 246 + 248 + 13 + 14 + 23 + 24 + 25 + 36 + 78$

8.37: Method 6 $(6, 2)$
$f = 2348 + 5678 + 128 + 138 + 145 + 146 + 148 + 178 + 238 + 247 + 258 + 268 + 347 + 348 + 567 + 568 + 578 + 14 + 17 + 25 + 26 + 27 + 35 + 37 + 38 + 46 + 48 + 56 + 67 + 68 + 6$

8.38: Dual of 8.37
$f = 1234 + 1567 + 123 + 137 + 147 + 157 + 158 + 234 + 235 + 245 + 278 + 367 + 467 + 12 + 15 + 24 + 25 + 26 + 28 + 35 + 36 + 38 + 48 + 67 + 78$

8.39: Method 6 $(6, 2)$
$f = 1234 + 1567 + 128 + 147 + 156 + 157 + 167 + 245 + 247 + 345 + 478 + 567 + 12 + 14 + 25 + 36 + 37 + 45 + 46 + 47 + 56 + 57 + 58 + 67 + 78$

8.40: Method 6 $(6, 2)$
$f = 2348 + 5678 + 168 + 234 + 248 + 256 + 257 + 258 + 267 + 268 + 348 + 357 + 358 + 367 + 468 + 568 + 678 + 12 + 13 + 23 + 25 + 26 + 35 + 37 + 38 + 46 + 48 + 58 + 3$

8.41: Dual of 8.41
$f = 1234 + 1567 + 125 + 128 + 134 + 137 + 138 + 156 + 167 + 178 + 245 + 247 + 478 + 567 + 12 + 14 + 17 + 25 + 37 + 46 + 56 + 57 + 58 + 67 + 78$

8.42: Method 6 $(6, 2)$
$f = 1358 + 3478 + 135 + 234 + 248 + 258 + 268 + 347 + 348 + 358 + 368 + 456 + 468 + 478 + 578 + 14 + 17 + 24 + 25 + 34 + 36 + 46 + 47 + 56 + 58$

8.43: Method 6 $(6, 2)$
$f = 1358 + 135 + 234 + 248 + 258 + 268 + 358 + 368 + 456 + 468 + 14 + 17 + 24 + 25 + 28 + 36 + 46 + 48 + 56$

8.44: Method 6 $(6, 2)$
$f = 1234 + 1256 + 1357 + 127 + 147 + 148 + 156 + 158 + 167 + 256 + 345 + 357 + 12 + 13 + 15 + 16 + 17 + 24 + 35 + 37 + 46 + 58 + 67$

8.45: Dual of 8.44
$f = 2468 + 3478 + 5678 + 168 + 178 + 234 + 238 + 246 + 256 + 267 + 268 + 348 + 356 + 367 + 568 + 578 + 678 + 12 + 13 + 16 + 18 + 23 + 25 + 28 + 35 + 36 + 37 + 48 + 67 + 78 + 2 + 6$

8.46: Method 6 $(6, 2)$
$f = 1234 + 1256 + 127 + 128 + 145 + 146 + 148 + 157 + 234 + 246 + 256 + 13 + 16 + 24 + 28 + 34 + 35 + 47$

8.47: Dual of 8.46
$f = 3478 + 5678 + 138 + 178 + 236 + 238 + 267 + 268 + 278 + 348 + 467 + 468 + 567 + 568 + 578 + 16 + 26 + 28 + 35 + 38 + 47 + 56 + 57 + 68$

8.48: Method 6 $(6, 2)$
$f = 2357 + 2468 + 3478 + 5678 + 178 + 234 + 235 + 237 + 238 + 246 + 247 + 248 + 256 + 258 + 278 + 357 + 367 + 478 + 678 + 12 + 24 + 25 + 26 + 27 + 28 + 34 + 35 + 36 + 37 + 58 + 67 + 68 + 78 + 2 + 3 + 8$

8.49: Dual of 8.48
$f = 1234 + 1256 + 1357 + 1468 + 124 + 125 + 134 + 138 + 147 + 157 + 167 + 256 + 458 + 468 + 12 + 13 + 14 + 15 + 16 + 25 + 34 + 48 + 56 + 57 + 58 + 67$

8.50: Method 6 $(6, 2)$
$f = 1234 + 1256 + 1357 + 127 + 128 + 134 + 137 + 138 + 145 + 146 + 147 + 157 + 168 + 237 + 256 + 367 + 12 + 16 + 17 + 18 + 26 + 28 + 34 + 36 + 37 + 56 + 57 + 68$

8.51: Dual of 8.50
$f = 2468+3478+5678+178+234+238+246+247+256+267+346+368+456+458+468+568+12+14+16+18+23+27+36+38+45+46+47+48+57+67+68+78+1+3+7$

8.52: Method 6 $(6, 2)$
$f = 1234 + 1256 + 1357 + 125 + 126 + 127 + 128 + 136 + 137 + 138 + 145 + 156 + 157 + 237 + 256 + 12 + 15 + 16 + 25 + 26 + 28 + 34 + 36 + 57$

8.53: Method 6 $(6, 2)$
$f = 2348 + 3478 + 4568 + 128 + 148 + 178 + 234 + 268 + 278 + 347 + 358 + 456 + 478 + 678 + 14 + 26 + 28 + 34 + 35 + 47 + 48 + 56 + 67 + 68 + 78$

8.54: Method 6 $(6, 2)$
$f = 1234 + 1256 + 1378 + 125 + 127 + 128 + 134 + 136 + 147 + 156 + 157 + 168 + 178 + 256 + 257 + 12 + 13 + 14 + 15 + 24 + 37 + 56 + 57 + 68 + 78$

8.55: Dual of 8.54
$f = 2456 + 3478 + 5678 + 148 + 245 + 248 + 258 + 357 + 378 + 456 + 457 + 458 + 567 + 568 + 578 + 15 + 18 + 24 + 28 + 36 + 37 + 48 + 56 + 57 + 58 + 68 + 8$

8.56: Method 6 $(6, 2)$
$f = 2468+3456+3478+5678+146+267+268+278+348+357+358+367+368+378+456+457+458+478+567+578+17+18+26+27+35+37+46+57+78+7$

8.57: Dual of 8.56
$f = 1234 + 1256 + 1278 + 1357 + 128 + 135 + 136 + 145 + 147 + 148 + 156 + 157 + 278 + 12 + 18 + 26 + 27 + 35 + 47 + 48$

8.58: Method 6 $(6, 2)$
$f = 1234 + 1256 + 1378 + 125 + 126 + 127 + 128 + 134 + 136 + 137 + 138 + 148 + 167 + 258 + 357 + 378 + 14 + 16 + 17 + 24 + 25 + 35 + 37 + 45 + 48 + 57 + 67 + 78$

8.59: Dual of 8.59
$f = 2456 + 3478 + 5678 + 134 + 156 + 168 + 235 + 236 + 238 + 245 + 248 + 278 + 345 + 347 + 358 + 368 + 378 + 458 + 567 + 568 + 678 + 12 + 13 + 16 + 18 + 23 + 24 + 26 + 27 + 28 + 37 + 38 + 48 + 56 + 2$

8.60: Method 6 $(6, 2)$

$f = 1234 + 1256 + 1357 + 127 + 136 + 137 + 147 + 148 + 156 + 157 + 248 + 258 + 578 + 12 + 15 + 16 + 17 + 23 + 26 + 37 + 38 + 45 + 47 + 48 + 58 + 68 + 78$

8.61: Dual of 8.60

$f = 2468 + 3478 + 5678 + 134 + 156 + 235 + 236 + 248 + 268 + 278 + 345 + 356 + 358 + 368 + 456 + 468 + 678 + 12 + 16 + 18 + 24 + 28 + 34 + 35 + 36 + 37 + 45 + 46 + 67 + 78 + 6 + 8$

8.62: Method 6 $(6, 2)$

$f = 1234 + 1256 + 1278 + 1357 + 135 + 137 + 138 + 147 + 158 + 167 + 178 + 234 + 245 + 248 + 345 + 12 + 15 + 18 + 23 + 24 + 25 + 26 + 28 + 34 + 35 + 37 + 38 + 46 + 48 + 57$

8.63: Dual of 8.62

$f = 2468 + 3456 + 3478 + 5678 + 136 + 156 + 178 + 234 + 236 + 237 + 246 + 257 + 267 + 278 + 345 + 347 + 356 + 367 + 457 + 467 + 468 + 478 + 567 + 578 + 12 + 13 + 14 + 15 + 25 + 35 + 37 + 38 + 45 + 56 + 58 + 78 + 2 + 3 + 4 + 6 + 8$

8.64: Method 6 $(6, 2)$

$f = 1234 + 1235 + 1236 + 1248 + 1357 + 1456 + 1567 + 1678 + 124 + 125 + 128 + 134 + 145 + 146 + 156 + 158 + 167 + 178 + 234 + 235 + 236 + 248 + 256 + 12 + 14 + 17 + 18 + 23 + 26 + 27 + 34 + 35 + 36 + 37 + 46 + 47 + 67 + 78 + 7$

8.65: Dual of 8.64

$f = 2345 + 2348 + 2378 + 2468 + 3567 + 4578 + 4678 + 5678 + 134 + 135 + 137 + 138 + 145 + 146 + 168 + 178 + 234 + 236 + 237 + 246 + 247 + 256 + 268 + 278 + 346 + 347 + 348 + 357 + 358 + 367 + 467 + 567 + 568 + 578 + 678 + 12 + 13 + 14 + 17 + 18 + 24 + 25 + 26 + 27 + 35 + 36 + 46 + 47 + 57 + 67 + 78 + 1 + 6$

8.66: Method 6 $(6, 2)$

$f = 2345 + 2346 + 2357 + 2467 + 2568 + 2578 + 3458 + 4678 + 5678 + 124 + 125 + 126 + 127 + 146 + 148 + 157 + 158 + 234 + 236 + 237 + 238 + 257 + 267 + 278 + 347 + 348 + 356 + 367 + 456 + 457 + 467 + 468 + 478 + 13 + 17 + 23 + 24 + 28 + 36 + 37 + 38 + 45 + 56 + 57 + 58 + 78 + 1 + 3 + 5 + 6 + 7 + 8$

8.67: Dual of 8.66

$f = 1234 + 1235 + 1267 + 1346 + 1347 + 1358 + 1468 + 1578 + 1678 + 124 + 134 + 136 + 138 + 156 + 178 + 234 + 235 + 346 + 347 + 358 + 15 + 16 + 17 + 18 + 23 + 24 + 28 + 35 + 36 + 46 + 47 + 48 + 58 + 67 + 78$

8.68: Method 6 $(4, 4)$

$f = 1234 + 1237 + 1238 + 1245 + 1247 + 1248 + 1267 + 1268 + 1278 + 1345 + 1348 + 1356 + 1357 + 1367 + 1456 + 1457 + 1458 + 1478 + 2346 + 2356 + 2357 + 2358 + 2367 + 2368 + 2456 + 2457 + 3458 + 3478 + 123 + 124 + 125 + 134 + 135 + 137 + 147 + 148 +$

$156+158+167+245+246+247+248+257+258+267+278+346+347+357+468+12+14+15+16+24+26+27+35+37+38+47+56+78+2+3$

8.69: Dual of 8.68
$f = 1256+1267+1368+1378+1457+1458+1467+1468+1478+1578+2356+2367+2368+2378+2458+2468+2478+2567+2678+3456+3457+3458+3567+3568+3678+4567+4568+5678+127+135+138+147+148+156+157+167+235+237+238+258+267+268+278+345+357+358+458+467+468+567+568+578+14+15+17+18+23+24+25+28+34+36+38+56+57+67+68+78$

8.70: Method 6 $(4,4)$
$f = 1234+1235+1258+1268+1346+1356+1358+1367+1457+1458+2348+2356+2378+2457+2468+3457+3467+124+138+145+146+156+158+178+236+245+247+248+257+258+267+345+346+356+357+358+367+378+456+457+467+13+14+15+17+23+25+27+36+37+57+68+78$

8.71: Method 6 $(4,4)$
$f = 1256+1258+1358+1368+1456+1458+1468+1568+1578+1678+2358+2367+2458+2467+2568+2578+3456+3458+3467+3468+3478+3567+3568+4578+128+135+136+145+146+148+157+158+167+168+237+238+245+247+256+258+267+347+357+358+378+457+678+12+13+15+17+23+24+26+27+28+35+36+37+45+46+58$

8.72: Method 6 $(4,4)$
$f = 1267+1356+1367+1368+1378+1457+1458+1568+1578+2357+2367+2378+2467+2567+3456+3457+3458+3467+3468+3478+3567+3578+3678+4578+4678+125+126+127+147+148+157+167+168+178+236+238+245+246+256+257+267+268+278+345+356+368+378+456+467+468+568+13+14+15+18+24+25+35+36+37+38+45+48+56+57+68+78$

8.73: Dual of 8.72
$f = 1235+1236+1245+1246+1248+1256+1257+1258+1267+1268+1278+1348+1358+1456+1458+1468+2346+2347+2367+2368+2456+2457+2458+2478+3458+123+124+127+128+134+135+137+138+146+156+157+167+168+235+237+267+268+345+347+357+378+456+458+467+468+478+12+13+18+23+25+26+35+45+47+48+57+58+67+78+2+3+4+8$

8.74: Method 6 $(4,4)$
$f = 1256+1257+1268+1278+1356+1367+1457+1458+1467+1567+1568+1678+2357+2367+2368+2378+2456+2458+2467+2468+2478+2578+3456+3457+3458+3567+3568+4567+4568+5678+125+126+127+128+137+145+$

$146 + 156 + 157 + 237 + 245 + 246 + 257 + 258 + 268 + 345 + 346 + 347 + 367 + 378 + 468 + 568 + 678 + 14 + 23 + 25 + 36 + 38 + 45 + 46 + 47 + 48 + 56 + 57 + 58 + 67 + 78$

8.75: Method 6 $(4, 4)$

$f = 1256 + 1257 + 1268 + 1278 + 1357 + 1367 + 1457 + 1458 + 1467 + 1578 + 1678 + 2356 + 2367 + 2368 + 2378 + 2456 + 2458 + 2467 + 2468 + 2478 + 2578 + 2678 + 3456 + 3457 + 3458 + 3578 + 4567 + 4568 + 126 + 127 + 136 + 145 + 146 + 157 + 158 + 167 + 168 + 178 + 236 + 245 + 246 + 256 + 258 + 345 + 346 + 347 + 356 + 368 + 468 + 12 + 14 + 15 + 16 + 18 + 23 + 26 + 27 + 28 + 38 + 45 + 46 + 47 + 48 + 67 + 68 + 78$

8.76: Dual of 8.75

$f = 1237+1238+1246+1267+1268+1278+1345+1346+1356+1357+1358+1367+1378+1456+1457+1458+1478+2345+2346+2358+2367+2368+2458+2468+3456+3457+3468+3478+125+134+138+145+146+147+157+158+167+178+234+235+236+237+238+257+267+278+347+348+368+456+457+458+467+468+26+27+28+34+35+36+37+38+45+46+47+48+56+58+67+2+3+4+5$

8.77: Method 6 $(4, 4)$

$f = 1238+1248+1346+1358+1367+1368+1458+1467+1468+2345+2346+2367+2378+2467+2478+3456+3457+3468+3478+127+136+138+146+147+157+234+236+237+238+248+267+278+345+346+348+356+367+378+457+467+468+12+13+14+16+17+23+25+27+28+34+35+38+45+46+47+48+56+58+78$

8.78: Dual of 8.77

$f = 1256 + 1257 + 1268 + 1278 + 1356 + 1358 + 1456 + 1458 + 1578 + 1678 + 2357 + 2358 + 2367 + 2457 + 2458 + 2467 + 2578 + 3567 + 4567 + 125 + 127 + 136 + 145 + 146 + 148 + 156 + 157 + 158 + 167 + 236 + 238 + 245 + 248 + 256 + 257 + 258 + 345 + 346 + 378 + 456 + 457 + 468 + 568 + 678 + 13 + 17 + 24 + 26 + 27 + 28 + 37 + 45 + 46 + 47 + 56 + 57 + 58 + 68 + 78 + 1 + 2 + 5 + 7$

8.79: Method 6 $(4, 4)$

$f = 1238 + 1248 + 1346 + 1347 + 1358 + 1367 + 1368 + 1458 + 1467 + 1468 + 2345 + 2346 + 2347 + 2367 + 2378 + 2467 + 2478 + 3456 + 3457 + 3468 + 3478 + 127 + 134 + 136 + 137 + 138 + 146 + 157 + 236 + 238 + 247 + 248 + 267 + 278 + 356 + 367 + 378 + 457 + 467 + 468 + 12 + 16 + 24 + 25 + 28 + 34 + 36 + 47 + 56 + 58 + 78$

8.80: Dual of 8.79

$f = 1256 + 1257 + 1268 + 1278 + 1356 + 1358 + 1456 + 1458 + 1568 + 1578 + 1678 + 2357 + 2358 + 2367 + 2457 + 2458 + 2467 + 2568 + 2578 + 3567 + 4567 + 125 + 126 + 127+128+138+145+167+178+235+236+267+345+346+356+378+457+468+14+16+17+18+23+27+35+36+37+38+45+46+47+48+56+57+58+3+4$

8.81: Method 6 $(4,4)$

$f = 1238+1248+1345+1346+1347+1358+1367+1368+1457+2348+2367+2378+2457+2458+2468+2478+3456+3467+3468+127+134+135+136+137+138+158+167+168+234+235+237+245+247+257+258+268+278+345+346+356+357+468+478+12+23+24+34+35+46+56+57+58+67$

8.82: Dual of 8.81

$f = 1257+1258+1278+1356+1357+1367+1368+1456+1458+1567+2368+2457+2458+2467+2568+2578+2678+3567+4567+126+137+138+157+167+178+235+236+238+245+246+247+345+346+356+357+358+367+456+468+678+12+14+15+16+17+18+24+28+34+35+45+46+47+57+78+4$

8.83: Method 6 $(4,4)$

$f = 1257+1258+1268+1278+1356+1358+1456+1458+2357+2358+2367+2457+2458+2467+2567+2678+3567+4567+125+126+127+135+145+146+148+156+158+167+168+178+235+236+237+238+247+248+258+267+268+345+346+357+378+456+468+568+578+12+13+15+16+34+45+46+47+48+56+57+58+67+78$

8.84: Dual of 8.83

$f = 1238+1248+1345+1348+1358+1367+1368+1458+1467+1468+2367+2378+2467+2478+3456+3457+3467+3468+123+124+127+128+135+137+157+158+167+168+234+235+236+237+238+245+247+346+357+358+367+378+457+458+12+17+18+24+25+28+35+38+45+46+47+56+78+2+3+4+5+6$

8.85: Method 6 $(4,4)$

$f = 1256+1258+1268+1278+1358+1378+1456+1457+1468+1568+1678+2356+2357+2358+2368+2456+2458+2468+2478+2568+2578+2678+3456+3458+3468+3568+3578+4567+4568+4578+4678+125+126+135+136+146+157+167+168+235+236+245+246+258+268+278+348+356+367+467+468+478+14+17+18+23+25+26+28+37+46+47+67+78$

8.86: Dual of 8.85

$f = 1235+1236+1237+1238+1246+1247+1257+1267+1278+1345+1346+1347+1356+1357+1367+1378+1457+1467+1468+1478+2345+2347+2357+2368+2378+2456+2467+3456+3457+3467+3478+124+125+126+128+134+135+136+138+146+147+157+158+167+168+178+235+236+237+238+245+246+247+258+267+347+357+368+378+457+458+467+468+478+13+14+15+16+23+24+25+26+28+34+36+38+47+56+67+78+4+5+6+8$

8.87: Method 6 $(4,4)$

$f = 1256+1258+1268+1278+1358+1378+1456+1457+1468+1568+1678+$

$2358 + 2456 + 2458 + 2468 + 2478 + 2678 + 3456 + 3458 + 3468 + 3568 + 3578 + 4567 + 4568 + 4578 + 4678 + 125 + 126 + 135 + 136 + 146 + 157 + 167 + 168 + 235 + 245 + 246 + 258 + 268 + 278 + 348 + 357 + 367 + 368 + 467 + 468 + 478 + 568 + 578 + 14 + 17 + 18 + 23 + 25 + 26 + 28 + 36 + 37 + 46 + 47 + 67 + 78 + 1 + 3 + 4 + 7 + 8$

8.88: Dual of 8.87
$f = 1235 + 1236 + 1237 + 1238 + 1246 + 1247 + 1257 + 1267 + 1278 + 1345 + 1356 + 1357 + 1367 + 1378 + 1467 + 2345 + 2347 + 2357 + 2368 + 2378 + 2456 + 2467 + 3456 + 3457 + 3467 + 3478 + 125 + 127 + 128 + 134 + 135 + 137 + 145 + 146 + 156 + 158 + 167 + 168 + 234 + 236 + 238 + 246 + 247 + 256 + 258 + 267 + 268 + 346 + 348 + 357 + 368 + 378 + 456 + 458 + 467 + 468 + 18 + 23 + 24 + 25 + 27 + 35 + 36 + 38 + 56 + 57 + 68 + 3 + 5 + 6 + 8$

8.89: Method 6 $(4, 4)$
$f = 1258 + 1278 + 1357 + 1358 + 1367 + 1378 + 1456 + 1467 + 1468 + 1568 + 2358 + 2378 + 2458 + 2478 + 2568 + 2578 + 2678 + 3456 + 3457 + 3458 + 3468 + 3478 + 3568 + 3578 + 4567 + 5678 + 125 + 127 + 128 + 135 + 136 + 138 + 145 + 147 + 148 + 157 + 245 + 247 + 248 + 256 + 257 + 267 + 268 + 345 + 346 + 348 + 356 + 357 + 358 + 367 + 368 + 378 + 457 + 458 + 467 + 468 + 478 + 567 + 568 + 578 + 678 + 14 + 15 + 16 + 18 + 23 + 26 + 27 + 34 + 36 + 37 + 38 + 45 + 46 + 47 + 57 + 68$

8.90: Dual of 8.89
$f = 1234 + 1238 + 1246 + 1247 + 1256 + 1257 + 1267 + 1268 + 1278 + 1345 + 1346 + 1347 + 1356 + 1367 + 1456 + 1467 + 2347 + 2357 + 2358 + 2378 + 2456 + 2458 + 2467 + 2468 + 3456 + 3467 + 123 + 124 + 125 + 126 + 127 + 128 + 137 + 138 + 145 + 147 + 148 + 156 + 157 + 158 + 236 + 237 + 245 + 246 + 247 + 248 + 257 + 258 + 267 + 268 + 278 + 345 + 346 + 347 + 348 + 356 + 357 + 368 + 456 + 458 + 467 + 468 + 13 + 14 + 16 + 23 + 27 + 34 + 37 + 46 + 58 + 67 + 78 + 3 + 5 + 6 + 8$

8.91: Method 6 $(4, 4)$
$f = 1258 + 1278 + 1357 + 1358 + 1367 + 1378 + 1456 + 1467 + 1468 + 1678 + 2458 + 2478 + 2568 + 2578 + 2678 + 3456 + 3457 + 3458 + 3468 + 3478 + 3567 + 4567 + 5678 + 125 + 127 + 128 + 136 + 137 + 145 + 147 + 148 + 156 + 157 + 167 + 168 + 245 + 247 + 248 + 256 + 257 + 258 + 267 + 268 + 278 + 345 + 346 + 348 + 358 + 378 + 457 + 458 + 467 + 468 + 478 + 568 + 13 + 14 + 17 + 23 + 26 + 27 + 34 + 37 + 38 + 45 + 46 + 47 + 68$

8.92: Method 6 $(4, 4)$
$f = 1256 + 1257 + 1258 + 1358 + 1456 + 1457 + 1567 + 1568 + 1578 + 2358 + 2458 + 2467 + 2678 + 3458 + 3467 + 3678 + 4567 + 4678 + 126 + 127 + 128 + 138 + 145 + 146 + 147 + 156 + 167 + 235 + 238 + 245 + 248 + 256 + 345 + 348 + 358 + 367 + 456 + 458 + 567 + 578 + 12 + 13 + 14 + 15 + 16 + 17 + 18 + 24 + 25 + 28 + 48 + 56 + 58 + 68$

8.93: Method 6 $(4,4)$
$f = 1256+1257+1258+1267+1358+1367+1456+1457+1568+1578+1678+2357+2358+2367+2368+2378+2458+2467+2568+2678+3458+3467+4567+4678+5678+126+127+128+135+138+145+146+147+167+235+236+237+238+245+248+257+267+268+278+345+348+357+367+368+378+456+458+578+678+12+13+14+16+17+18+24+25+26+27+28+35+36+37+48+56+57+78$

8.94: Dual of 8.93
$f = 1234+1235+1238+1258+1267+1345+1347+1358+1367+1456+1457+1458+1467+1468+2345+2346+2347+2368+2378+2458+2467+3458+3467+3468+3478+123+126+127+128+134+146+147+156+157+158+167+168+234+235+236+237+245+246+247+248+256+257+268+278+348+378+456+457+458+468+478+12+15+16+23+24+27+34+37+38+45+46+47+48+57+68+2+4+5+8$

8.95: Method 6 $(4,4)$
$f = 1256+1257+1267+1367+1456+1457+1567+1678+2367+2458+2467+3458+3467+3567+3568+3578+3678+4567+4678+125+126+127+136+137+145+146+147+156+167+168+178+245+248+256+345+348+356+357+358+456+458+568+678+14+18+23+24+25+28+38+48+57+68$

8.96: Dual of 8.95
$f = 1235+1238+1245+1246+1247+1248+1258+1267+1358+1367+1458+2345+2348+2368+2378+2458+3458+3468+3478+126+127+128+135+136+146+147+167+234+235+238+245+248+256+257+267+348+367+378+456+457+13+23+27+34+36+45+46+47+57+67+68+78+6$

8.97: Method 6 $(4,4)$
$f = 1238+1358+1367+1368+2348+2367+2378+3458+3467+3468+123+127+134+138+157+235+236+238+247+267+278+345+346+347+348+356+358+368+378+457+18+23+28+38+47+48+56+57+58+78$

8.98: Dual of 8.97
$f = 1257+1258+1267+1456+1458+1567+2457+2458+2467+4567+125+126+135+136+145+156+158+167+245+247+256+257+268+345+346+456+457+467+468+14+15+17+18+23+26+34+46+56+1$

8.99: Method 6 $(4,4)$
$f = 1238+1345+1346+1347+1348+1358+1367+1368+2345+2346+2347+2367+2378+3457+3458+3468+3478+123+127+138+145+146+147+148+157+234+235+236+238+245+246+248+267+278+356+358+368+378+467+478+14+18+23+24+28+34+38+45+46+56+57+58+78$

8.100: Dual of 8.99
$f = 1256 + 1257 + 1267 + 1268 + 1456 + 1458 + 1568 + 1578 + 1678 + 2457 + 2458 + 2467 + 2567 + 2568 + 2578 + 2678 + 4567 + 125 + 135 + 136 + 145 + 156 + 168 + 245 + 247 + 258 + 345 + 346 + 456 + 457 + 467 + 468 + 568 + 578 + 678 + 14 + 16 + 17 + 18 + 23 + 25 + 27 + 34 + 35 + 36 + 46 + 56 + 57 + 3$

8.101: Method 6 $(4, 4)$
$f = 1238 + 1347 + 1358 + 1367 + 1368 + 2347 + 2348 + 2367 + 2378 + 3458 + 3467 + 3468 + 123 + 127 + 134 + 138 + 147 + 157 + 235 + 236 + 267 + 278 + 356 + 367 + 378 + 457 + 13 + 18 + 23 + 27 + 28 + 34 + 35 + 36 + 38 + 45 + 46 + 56 + 58 + 78$

8.102: Dual of 8.101
$f = 1257 + 1258 + 1267 + 1456 + 1458 + 1567 + 1568 + 2457 + 2458 + 2467 + 2568 + 4567 + 126 + 135 + 136 + 156 + 157 + 168 + 245 + 246 + 247 + 256 + 257 + 258 + 345 + 346 + 457 + 467 + 468 + 567 + 568 + 14 + 15 + 17 + 18 + 23 + 24 + 25 + 27 + 34 + 35 + 36 + 45 + 46 + 56 + 2 + 3 + 4 + 5 + 6$

8.103: Method 6 $(4, 4)$
$f = 1238 + 1358 + 1367 + 1368 + 2367 + 2378 + 2478 + 3457 + 3467 + 3468 + 3478 + 123 + 127 + 134 + 138 + 157 + 234 + 235 + 236 + 238 + 267 + 278 + 345 + 346 + 356 + 358 + 368 + 378 + 458 + 18 + 23 + 28 + 34 + 38 + 47 + 48 + 56 + 57 + 58 + 78$

8.104: Dual of 8.103
$f = 1256 + 1257 + 1258 + 1268 + 1356 + 1456 + 1458 + 2457 + 2458 + 2467 + 4567 + 125 + 126 + 135 + 136 + 145 + 158 + 167 + 245 + 247 + 256 + 257 + 268 + 345 + 346 + 456 + 457 + 467 + 468 + 14 + 15 + 17 + 18 + 23 + 26 + 34 + 46 + 56 + 1$

8.105: Method 6 $(4, 4)$
$f = 1256 + 1257 + 1267 + 1456 + 1458 + 1567 + 1568 + 1578 + 1678 + 2457 + 2458 + 2467 + 2567 + 2568 + 2578 + 2678 + 4567 + 125 + 126 + 127 + 145 + 156 + 246 + 247 + 256 + 257 + 258 + 267 + 268 + 345 + 346 + 457 + 467 + 468 + 567 + 568 + 578 + 678 + 12 + 13 + 14 + 17 + 18 + 23 + 24 + 25 + 26 + 34 + 45 + 56 + 58 + 67$

8.106: Dual of 8.105
$f = 1238 + 1345 + 1346 + 1347 + 1348 + 1358 + 1367 + 1368 + 2345 + 2346 + 2347 + 2348 + 2367 + 2378 + 3458 + 3468 + 3478 + 123 + 127 + 138 + 145 + 146 + 147 + 148 + 157 + 235 + 236 + 245 + 246 + 247 + 248 + 267 + 278 + 347 + 356 + 367 + 378 + 457 + 467 + 478 + 14 + 18 + 24 + 27 + 28 + 34 + 45 + 46 + 48 + 56 + 58 + 78 + 4 + 7 + 8$

8.107: Method 6 $(4, 4)$
$f = 1257 + 1258 + 1267 + 1268 + 1456 + 1458 + 2457 + 2458 + 2467 + 4567 + 125 +$

$127+145+156+158+167+246+247+256+257+268+345+346+457+467+468+12+13+14+17+18+23+24+25+26+34+45+56$

8.108: Dual of 8.107
$f = 1238+1358+1367+1368+2367+2378+3457+3458+3467+3468+123+127+134+138+157+234+235+236+267+278+345+346+348+356+367+378+18+27+28+47+56+58+78+7+8$

8.109: Method 6 $(4,4)$
$f = 1238+1247+1248+1345+1346+1347+1348+1358+1367+1368+1457+1458+1468+2345+2346+2347+2367+2378+2467+2478+3457+3478+123+124+127+146+157+234+235+236+238+245+247+267+278+356+358+367+368+457+467+478+13+17+18+23+24+28+34+35+36+46+56+57+58+67$

8.110: Dual of 8.109
$f = 1256+1268+1356+1358+1456+1458+1568+1578+1678+2357+2367+2368+2457+2458+2467+2567+2568+2578+2678+3567+3568+4567+125+127+128+157+158+235+236+237+238+247+267+278+345+346+356+357+367+368+456+457+467+468+567+678+12+14+15+17+18+25+26+27+28+34+45+46+56+57+68+2+4$

8.111: Method 6 $(4,4)$
$f = 1257+1258+1267+1356+1358+1456+1458+1567+1578+1678+2357+2367+2368+2457+2458+2467+2578+2678+3567+3568+4567+125+127+128+135+145+156+158+236+237+238+246+247+257+267+278+345+346+356+357+367+368+457+467+468+568+678+12+14+15+17+18+24+27+28+34+35+45+56+68$

8.112: Dual of 8.111
$f = 1238+1247+1248+1345+1346+1358+1367+1368+1457+1458+1468+2345+2346+2348+2367+2378+2467+2478+3458+3467+3468+123+124+127+146+147+148+157+234+235+236+238+245+248+267+278+348+356+358+367+368+458+468+13+17+18+23+24+28+34+35+36+46+48+56+57+58+67+3+7+8$

Appendix B. Stabiliser space dimensions $\underline{d}(f)$

Entries guaranteed to be 0 by Theorem 8.11 are blank. Functions in the same box (e.g. 8.16 and 8.17) are duals. The last column contains "No" if Theorem 8.13 shows that the class cannot be constructed by the Maiorana construction.

Class	d_1	d_2	d_3	d_4	d_5	d_6	d_7	Maiorana?
4.1	0		15					
6.1	0	0	0		63			
6.2	0	0	56		7			
6.3	32	0	30		1			
6.4	56	0	7		0			
8.1	0	0	0	0	0		255	
8.2	0	0	0	0	224		31	
8.3	0	0	0	240	0		15	
8.4	0	0	128	0	120		7	
8.5	0	0	192	48	8		7	
8.6	0	0	224	0	28		3	
8.7	0	128	100	16	8		3	
8.8	0	160	80	8	4		3	
8.9	0	192	24	36	0		3	
8.10	64	136	40	8	6		1	
8.11	96	136	20	2	0		1	No
8.12	128	112	12	2	0		1	No
8.13	128	112	0	14	0		1	
8.14	128	120	6	0	0		1	No
8.15	128	64	48	8	6		1	
8.16	64	112	60	8	10		1	
8.17	128	56	48	16	6		1	
8.18	128	40	72	8	6		1	
8.19	128	80	28	16	2		1	
8.20	128	64	52	8	2		1	No
8.21	128	68	52	4	2		1	No
8.22	128	72	40	12	2		1	
8.23	128	80	40	4	2		1	No
8.24	128	56	42	28	0		1	

Class	d_1	d_2	d_3	d_4	d_5	d_6	d_7	Maiorana?
8.25	128	72	42	12	0		1	No
8.26	128	80	42	4	0		1	No
8.27	96	132	26	0	0		1	No
8.28	120	132	2	0	0		1	No
8.29	120	126	8	0	0		1	No
8.30	128	126	0	0	0		1	No
8.31	120	102	32	0	0		1	No
8.32	128	114	12	0	0		1	No
8.33	96	120	24	12	2		1	
8.34	128	96	16	12	2		1	
8.35	96	144	0	12	2		1	
8.36	48	168	20	12	6		1	
8.37	226	26	2	1	0		0	No
8.38	228	23	3	1	0		0	No
8.39	218	32	4	1	0		0	No
8.40	212	28	12	2	1		0	No
8.41	220	27	5	3	0		0	No
8.42	220	28	5	2	0		0	No
8.43	84	154	14	2	0		1	No
8.44	196	42	13	3	1		0	No
8.45	221	11	18	4	1		0	No
8.46	160	72	8	12	3		0	
8.47	224	0	16	12	3		0	
8.48	224	24	6	0	1		0	No
8.49	237	11	6	1	0		0	No
8.50	216	26	11	1	1		0	No
8.51	221	19	12	2	1		0	No
8.52	224	16	0	14	1		0	
8.53	128	64	40	20	2		1	
8.54	220	27	5	3	0		0	No
8.55	224	16	12	2	1		0	No
8.56	224	16	12	2	1		0	No
8.57	228	19	6	2	0		0	No

Class	d_1	d_2	d_3	d_4	d_5	d_6	d_7	Maiorana?
8.58	240	12	3	0	0		0	No
8.59	248	6	1	0	0		0	No
8.60	241	12	2	0	0		0	No
8.61	248	6	1	0	0		0	No
8.62	245	9	1	0	0		0	No
8.63	248	6	1	0	0		0	No
8.64	246	9	0	0	0		0	No
8.65	248	6	1	0	0		0	No
8.66	248	6	1	0	0		0	No
8.67	249	6	0	0	0		0	No
8.68	252	3	0	0	0		0	No
8.69	255	0	0	0	0		0	No
8.70	255	0	0	0	0		0	No
8.71	252	3	0	0	0		0	No
8.72	249	6	0	0	0		0	No
8.73	255	0	0	0	0		0	No
8.74	248	6	1	0	0		0	No
8.75	248	6	1	0	0		0	No
8.76	252	3	0	0	0		0	No
8.77	236	18	1	0	0		0	No
8.78	242	12	1	0	0		0	No
8.79	236	18	1	0	0		0	No
8.80	248	6	1	0	0		0	No
8.81	246	9	0	0	0		0	No
8.82	248	0	7	0	0		0	No
8.83	248	6	1	0	0		0	No
8.84	255	0	0	0	0		0	No
8.85	240	8	6	1	0		0	No
8.86	244	9	2	0	0		0	No
8.87	249	6	0	0	0		0	No
8.88	252	3	0	0	0		0	No
8.89	244	9	2	0	0		0	No
8.90	248	6	1	0	0		0	No

Class	d_1	d_2	d_3	d_4	d_5	d_6	d_7	Maiorana?
8.91	240	14	0	1	0		0	No
8.92	249	6	0	0	0		0	No
8.93	246	9	0	0	0		0	No
8.94	252	3	0	0	0		0	No
8.95	242	12	1	0	0		0	No
8.96	248	6	1	0	0		0	No
8.97	200	38	12	4	1		0	No
8.98	224	8	18	4	1		0	No
8.99	220	27	5	3	0		0	No
8.100	224	16	12	2	1		0	No
8.101	200	38	12	4	1		0	No
8.102	216	16	17	5	1		0	No
8.103	241	12	2	0	0		0	No
8.104	248	0	7	0	0		0	No
8.105	216	32	5	1	1		0	No
8.106	224	24	4	3	0		0	No
8.107	224	18	12	0	1		0	No
8.108	229	17	8	1	0		0	No
8.109	245	9	1	0	0		0	No
8.110	248	0	7	0	0		0	No
8.111	245	3	7	0	0		0	No
8.112	245	9	1	0	0		0	No

Appendix C. Design automorphism and orbit details

Design orbit sizes are shown with multiplicities, so for example class 8.20 has four orbits, all of order 64.

Class(es)	$\lvert A\rvert$			$\lvert T\rvert$	$\lvert\operatorname{Aut}(\mathcal{A}(f))\rvert$	Orbit sizes
2.1	6	=	2.3	4	24	4
4.1	720	=	$2^4.3^2.5$	16	11,520	16
6.1	1,451,520	=	$2^9.3^4.5.7$	64	92,897,280	64
6.2	86,016	=	$2^{12}.3.7$	8	688,128	64
6.3	61,440	=	$2^{12}.3.5$	2	122,880	64
6.4	43,008	=	$2^{11}.3.7$	1	43,008	64
8.5	393,216	=	$2^{17}.3$	8	3,145,728	$192; 64$
8.7	16,384	=	2^{14}	4	65,536	
8.10	4,096	=	2^{12}	2	8,192	256
8.16 and 8.17	8,192	=	2^{13}	2	16,384	256
8.20	512	=	2^9	2	1,024	64^4
8.21	512	=	2^9	2	1,024	64^4
8.32	64	=	2^6	2	128	
8.33 and 8.34	2,048	=	2^{11}	2	4,096	128^2
8.36	1,536	=	$2^9.3$	2	3,072	$16^2; 32; 96^2$
8.37 and 8.38	16	=	2^4	1	16	16^{16}
8.39	16	=	2^4	1	16	16^{16}
8.40 and 8.41	32	=	2^5	1	32	$16^8; 32^4$
8.42	64	=	2^6	1	64	64^4
8.43	672	=	$2^5.3.7$	2	1,344	$32; 224$
8.44 and 8.45	64	=	2^6	1	64	64^4
8.46 and 8.47	1,024	=	2^{10}	1	1,024	32^8
8.48 and 8.49	4	=	2^2	1	4	$2^{32}; 4^{48}$
8.50 and 8.51	64	=	2^6	1	64	64^4
8.52	$\geqslant 4$	=	2^2	2	$\geqslant 8$	
8.53	8,192	=	2^{13}	2	16,384	$32^4; 128$
8.54 and 8.55	8	=	2^3	1	8	$4^{48}; 8^8$
8.56 and 8.57	$\geqslant 4$	=	2^2	1	$\geqslant 4$	
8.58 and 8.59	$\geqslant 2$	=	2	1	$\geqslant 2$	

Class(es)	$\lvert A\rvert$	$\lvert T\rvert$	$\lvert \operatorname{Aut}(\mathcal{A}(f))\rvert$	Orbit sizes
8.60 and 8.61	$32 = 2^5$	1	32	32^8
8.62 and 8.63	$32 = 2^5$	1	32	32^8
8.64 and 8.65	$4 = 2^2$	1	4	4^{64}
8.66 and 8.67	$\geqslant 2 = 2$	1	$\geqslant 2$	
8.70	$1 = 1$	1	1	1^{256}
8.74	$\geqslant 2 = 2$	1	$\geqslant 2$	
8.77 and 8.78	$16 = 2^4$	1	16	
8.79 and 8.80	$16 = 2^4$	1	16	
8.85 and 8.86	$\geqslant 2 = 2$	1	$\geqslant 2$	
8.89 and 8.90	$\geqslant 2 = 2$	1	$\geqslant 2$	
8.92	$4 = 2^2$	1	4	
8.93 and 8.94	$4 = 2^2$	1	4	
8.95 and 8.96	$2 = 2$	1	2	

References

1. A. Barlotti, Finite geometries and designs, *Surveys in Combinatorics 1987*, ed. C. Whitehead, *LMS Lecture Note Series* **123**, Cambridge Univ. Press, 1987, 1–12.
2. T. S. Blyth and E. F. Robertson, *Groups*, Chapman and Hall, 1986.
3. B. Bollobás, *Combinatorics*, Cambridge Univ. Press, 1986.
4. R. C. Bose, Strongly-regular graphs, partial geometries and partially balanced designs, *Pacific J. Math.* **13**, 1963, 389–419.
5. A. E. Brouwer, A. M. Cohen and A. Neumaier, *Distance-regular graphs*, Springer-Verlag, 1989.
6. R. Calderbank and W. M. Kantor, The geometry of two-weight codes, *Bulletin of the London Math. Soc.* **18**, 1986, 97–122.
7. P. J. Cameron and J. H. van Lint, *Designs, Graphs, Codes and their Links, LMS Student Text* **22**, Cambridge Univ. Press, 1991.
8. A. R. Camina, *A survey of the automorphism groups of block designs*, Manuscript, 1993.
9. W. Conley, *Optimisation: a simplified approach*, Petrocelli, 1981.
10. L. E. Dickson, *Linear Groups*, Dover, 1958.
11. D. M. Greig, *Optimisation*, Longman, 1980.
12. R. Hill, *A First Course in Coding Theory*, Oxford University Press, 1986.
13. J. W. P. Hirschfeld, *Projective geometries over finite fields*, Oxford Univ. Press, 1979.
14. J. W. P. Hirschfeld, Maximum sets in finite projective spaces, *Surveys in Combinatorics 1983*, ed. E. K. Lloyd, *LMS Lecture Note Series* **82**, Cambridge Univ. Press, 1983, 55–76.
15. A. J. Hoffman, On the uniqueness of the triangular association scheme, *Annals of Math. Statist.* **31**, 1960, 492–497.
16. D. Jungnickel and V. D. Tonchev, On Symmetric and Quasi-Symmetric Designs with the Symmetric Difference Property and Their Codes, *J. Comb. Theory* (A) **59**, 1992, 40–50.
17. W. M. Kantor, Symplectic Groups, Symmetric Designs, and Line Ovals, *J. Algebra* **33**, 1975, 43–58.

18. W. M. Kantor, 2-transitive designs, *Combinatorics*, ed. M. Hall Jr. and J. H. van Lint, Reidel, 1975, 365–418.

19. W. M. Kantor, *Automorphisms and isomorphisms of symmetric and affine designs*, Manuscript, 1992.

20. A. A. Kirillov, *Elements of the theory of representations*, Springer-Verlag, 1976.

21. P. Kleidman and M. Liebeck, *The Subgroup Structure of the Finite Classical Groups, LMS Lecture Note Series* **129**, Cambridge Univ. Press, 1990.

22. P. V. Kumar and R. A. Scholtz, Bounds on the Linear Span of Bent Sequences, *IEEE Transactions on Information Theory* IT-29 No. 6, November 1983, 854–862.

23. P. V. Kumar, R. A. Scholtz and L. R. Welch, Generalized Bent Functions and Their Properties, *J. Comb. Theory* (A) **40**, 1985, 90–107.

24. E. S. Lander, *Symmetric Designs: An Algebraic Approach, LMS Lecture Note Series* **74**, Cambridge Univ. Press, 1983.

25. Lehrstuhl D für Mathematik, *GAP: Groups, Algorithms and Programming*, RWTH Aachen, 1992.

26. A. Lempel and M. Cohn, Maximal Families of Bent Sequences, *IEEE Transactions on Information Theory* IT-28 No. 6, November 1982, 865–868.

27. R. L. McFarland, A Family of Difference Sets in Non-cyclic Groups, *J. Comb. Theory* (A) **15**, 1973, 1–10.

28. F. J. MacWilliams and N. J. Sloane, *The Theory of Error-correcting codes*, North-Holland, 1977.

29. Y. I. Manin, *Cubic forms: algebra, geometry, arithmetic*, North-Holland, 1974.

30. W. Meier and O. Staffelbach, Nonlinearity criteria for cryptographic functions, *Advances in Cryptology, Eurocrypt '89, Lecture Notes in Computer Science* **434**, 1989, 549–563.

31. K. Nyberg, Constructions of bent functions and difference sets, *Advances in Cryptology, Eurocrypt '90, Lecture Notes in Computer Science* **473**, 1990, 151–160.

32. J. Olsen, R. Scholtz and L. Welch, Bent-Function Sequences, *IEEE Transactions on Information Theory* IT-28 No. 6, November 1982, 858–864.

33. C. Parker, E. Spence and V. D. Tonchev, *Designs with the symmetric difference property on 64 points and their groups*, Manuscript, 1993.

34. B. Preneel, W. Van Leekwijck, L. Van Linden, R. Govaerts and J. Vandewalle, Propogation Characteristics of Boolean Functions, *Advances in Cryptology, Eurocrypt '90, Lecture Notes in Computer Science* **473**, 1990, 161–173.

35. O. S. Rothaus, On "Bent" Functions, *J. Comb. Theory* (A) **20**, 1976, 300–305.

36. M. S. Shrikhande and S. S. Sane, *Quasi-Symmetric Designs, LMS Lecture Note Series* **164**, Cambridge Univ. Press, 1992.

37. L. H. Soicher, GRAPE: a system for computing with graphs and groups, *Groups and Computation*, ed. L. Finkelstein and W. M. Kantor, *DIMACS Series in Discrete Mathematics and Theoretical Computer Science* **11**, Amer. Math. Soc., 1993, 287–291.

38. W. D. Wallis, Configurations Arising from Maximal Arcs, *J. Comb. Theory* (A) **14**, 1973, 115–119.

39. D. Welsh, *Codes and Cryptography*, Oxford Univ. Press, 1988.

40. R. Yarlagadda and J. E. Hershey, Analysis and synthesis of bent sequences, *IEEE Proceedings* (Part E) **136**, March 1989, 112–123.